`apa.cls`: A genuine LaTeX solution for psychological research articles

Athanassios Protopapas

Institute for Language & Speech Processing
Maroussi, Greece
E-mail: `protopap@ilsp.gr`

Psychological research manuscripts must usually conform to the guidelines of the American Psychological Association (APA) publication manual. LaTeX document class `apa.cls` implements the structural requirements of the manual, so that authors only have to concern themselves with manuscript content. By separating appearance from content, in LaTeX fashion, `apa.cls` can provide visually distinct outputs from the same manuscript file, thus producing manuscript-format or journal-style documents by switching a processing option. This article presents a bit of history and context for the development of `apa.cls`, noting the critical importance of an active online community of developers and users. There are several technical issues involved in handling the requirements of the APA manual, and these are discussed here along with their solution provided in `apa.cls`. The special macros and options of `apa.cls` are presented, with examples, on the topics of titles/headers, sectioning, lists, floats, typefaces, appendices, internationalization, and conditionals.

1 Introduction and goals

In the field of psychology, authors of scientific articles are typically required to submit their manuscripts for publication following the guidelines of the Publication Manual of the American Psychological Association (APA; Fifth Edition 2001, henceforth simply "the manual"). This set of guidelines is known as "APA style" (see also `apastyle.org`). The prescriptions of the manual are not limited to the usual requirements for double spacing and wide margins (meant to facilitate the editorial process as it was done in past decades, prior to the widespread adoption of electronic submission and computer processing), but also affect many, if not all, possible aspects of formatting, content structuring, vocabulary, and phrasing.

Naturally, what an author writes is not something that the computer can help with, as long as s/he has something to say. How an author formats the document, however, is not something that the author should have to deal with,

at any point during preparation of the manuscript. In contrast to the users of "word processing" software monsters, readers of this journal are familiar with the obvious fact that the appearance of a document is separable from, and largely independent of, the document's content. Moreover, technology allows most of the formatting concerns to be handled automatically, thus freeing the author to consider issues of content and structure. In particular, LaTeX is an extremely valuable tool built around this fundamental concept.

At a first approach, then, the existence of a set of guidelines for formatting manuscripts presents an obvious challenge to the LaTeX user. This challenge makes up the initial requirement for an APA-style processing option within the context of LaTeX. Because the manual defines precisely the form of every page of the manuscript, and not simply how to handle individual issues within a generic text form, the desired processing option must be a LaTeX *class* and not a style package. Using a document class, LaTeX is assigned the task of handling the appearance of the entire, complete document.

Beyond this first typesetting goal, which concerns submitting manuscripts to journal editors, the manual presents an additional problem for the APA-style writer. Specifically, by double-spacing the entire text and removing tables, figures, and footnotes from the point in the text where they are relevant, the manual ensures that the resulting manuscript is so highly unreadable and aesthetically offensive that it is not appropriate for dissemination. Taking into account the often extremely long delay between submission and appearance in print, this can be a serious impediment to the timely spread of useful scientific information.

Thus the goal of a comprehensive approach to the APA style within the scope of LaTeX must include an option to present a reasonably typeset article with easy to follow text and properly placed floating elements. Because LaTeX facilitates, indeed enforces, the separation of form from content, it is in fact possible to create a number of visually distinct forms of the same content, based on different typesetting options. This is what `apa.cls` accomplishes for the APA-style writer.

In summary, the main set of constraints that have guided the development of `apa.cls` are:

- For the production of manuscripts for submission to journal editors, full conformance to the specifications of the manual to the extent that can be handled automatically.
- Alternative document form familiar and easily readable for the audience of technical psychological literature.
- No effort on the part of the author related to the appearance of the manuscript or of the alternative form(s).
- Conformance to the standard LaTeX conventions and macros for common elements such as sectioning, captions etc.

- Minimal, and intuitive, extension of the standard macro set for APA-specific usage, aiming to support the LaTeX user as much as the APA-style author.

In the following sections I describe how these principles were applied in the development of `apa.cls` and I show the extent to which the stated goals for the document class have been achieved.

2 Overview and context of development

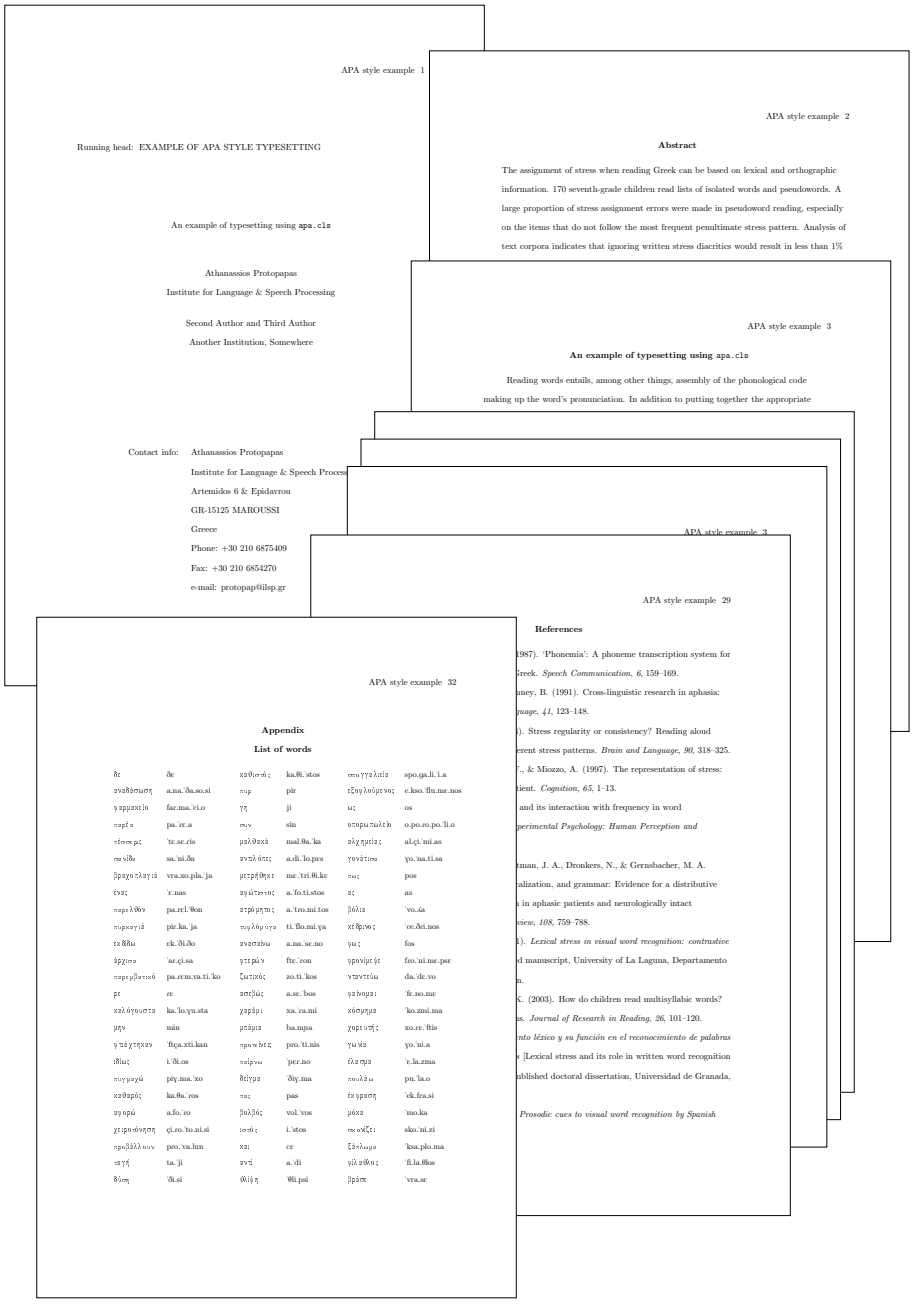
2.1 The relevance of historical context

The manual specifies precisely what each of the manuscript pages must contain. For example, manuscript page 1 (the "title page") must contain the page header and page number at the top right corner, a capitalized short title following the label "Running head:", the manuscript title, the author names grouped by institution followed by the corresponding affiliation(s) in a separate line, and so on. Figure 1 shows a few pages from a manuscript conforming to these rules. Note the separate pages for "Abstract," "Author Note," "Footnotes," "Figure Captions," as well as the table and figure pages.

Each of the manual specifications poses a separate problem. Some of these problems are relatively simple, such as setting the margins, headers, line spacing, (no) hyphenation etc. These can be addressed with standard LaTeX options, macros, or packages. At the time the first version of the predecessors of `apa.cls` was created, more than ten years ago, before the advent of LaTeX2e and the now more or less standard distrubutions, these issues were already easy to address with TeX commands such as for length setting. Today things are even easier, thanks to improvements and additions such as the `geometry` package, `fancyhdr`, `spacing`, etc. APA style, however, prescribed a number of much more complicated requirements that were not so straightforward to address with parameter setting approaches. These include sectioning, captions, floats, citations, footnotes, and a few others.

Fortunately, `apa.cls` was not created in a void. A number of LaTeX contributors had already worked on some of the most important components of APA requirements, including citations and sectioning. Among style files such as `theapa`, `newapa`, and `newapa2`, the approach of Young U. Ryu was the most complete and up-to-date at the time, and also addressed APA guidelines for lists, which deviate significantly from the LaTeX standard. Other packages, not tailored specifically to APA style but serving related purposes for other types of documents, were also available in CTAN. Among those, `endfloat` and `endnote` were probably the least dispensable, while `fancyheader` (later renamed `fancyhdr`) provided useful additional flexibility.

Perhaps the most severe APA-specific obstacle to the creation of a fully functional `apa.cls` was a fully conforming and well-behaving bibliography style to handle both in-text citations and document-end reference lists. This problem



δε	de	καθιστός	ka.θi.'stos	συνηγγελίδη	spo.ga.li.'i.a
αναβάσπωση	a.na.'βa.so.si	πυρ	pir	εξομολούμενος	e.kso.'fu.me.nos
γαρμαχίο	far.ma.'ci.o	γη	ji	ως	os
παρέα	pa.'re.a	σιν	sin	ατομπωλέθη	o.po.ro.po.'li.o
τέσσερις	'te.se.ris	μαλθακά	mal.θa.'ka	αλγηρίας	al.ɣi.'mi.as
σωλίδα	sa.'ni.ða	αντλότες	a.dli.'lo.pes	γονάτιση	ɣo.'na.ti.sa
βραχυπλαγιά	vra.xo.pla.'ja	μετρήθηκε	me.'tri.θi.ke	πως	pos
ένας	'e.nas	αγώνστους	a.'fo.ti.stos	ας	as
παρελθόν	pa.rel.'θon	στρόμησις	a.'tro.mi.tos	βόλιπ	'vo.ʎa
πυρκαγιά	pir.ka.'ja	πυλόμυγα	ti.'flo.mi.ɣa	κεθρινος	'ce.θri.nos
εκθέω	ek.'θi.ðo	αναπαδυ	a.na.'se.no	ψυς	fos
άρχισα	'ar.çi.sa	γτεράν	ftr.'con	προνίμής	fro.'ni.me.pse
παρεμβατικό	pa.rem.va.ti.'ko	ζωτικός	zo.ti.'kos	νταντούμ	da.'dr.vo
ρε	re	αστεβώς	a.se.'bos	φαδουμε	'fe.no.me
καλόγουστα	ka.'lo.yu.sta	χαρδιμ	xa.'ra.mi	κόσμημα	'ko.zmi.ma
μην	min	μπάμια	ba.mja	χορτυπής	xo.re.'ftis
γιάχτηκαν	'fiça.xti.kan	προωθητές	pro.'ti.nis	γωνία	yo.'ni.a
ίδιως	i.'ði.os	παίρνω	'per.no	έλασμα	'e.la.zma
συγμχώ	pïy.ma.'xo	δίγμα	'ðiy.ma	πουλάω	pu.'la.o
καθαρός	ka.θa.'ros	πας	pas	έκρασπ	'ek.fra.si
αγορώ	a.fo.'ro	βολβός	vol.'vos	μόχα	'mo.ka
χειρωνήσπ	çi.ro.'to.ni.si	ισσός	i.'stos	σωνάζει	sko.'ni.zi
προβδάλιων	pro.'va.lun	και	ce	ξάπλωμα	'ksa.plo.ma
ταγή	ta.'ji	αντί	a.'di	ηλιάθλος	'fi.la.θlos
δόση	'ði.si	θλίψη	'θli.psi	βράσπ	'vra.sc

Figure 1: Sample pages from an APA-style manuscript as can be submitted for publication to a journal.

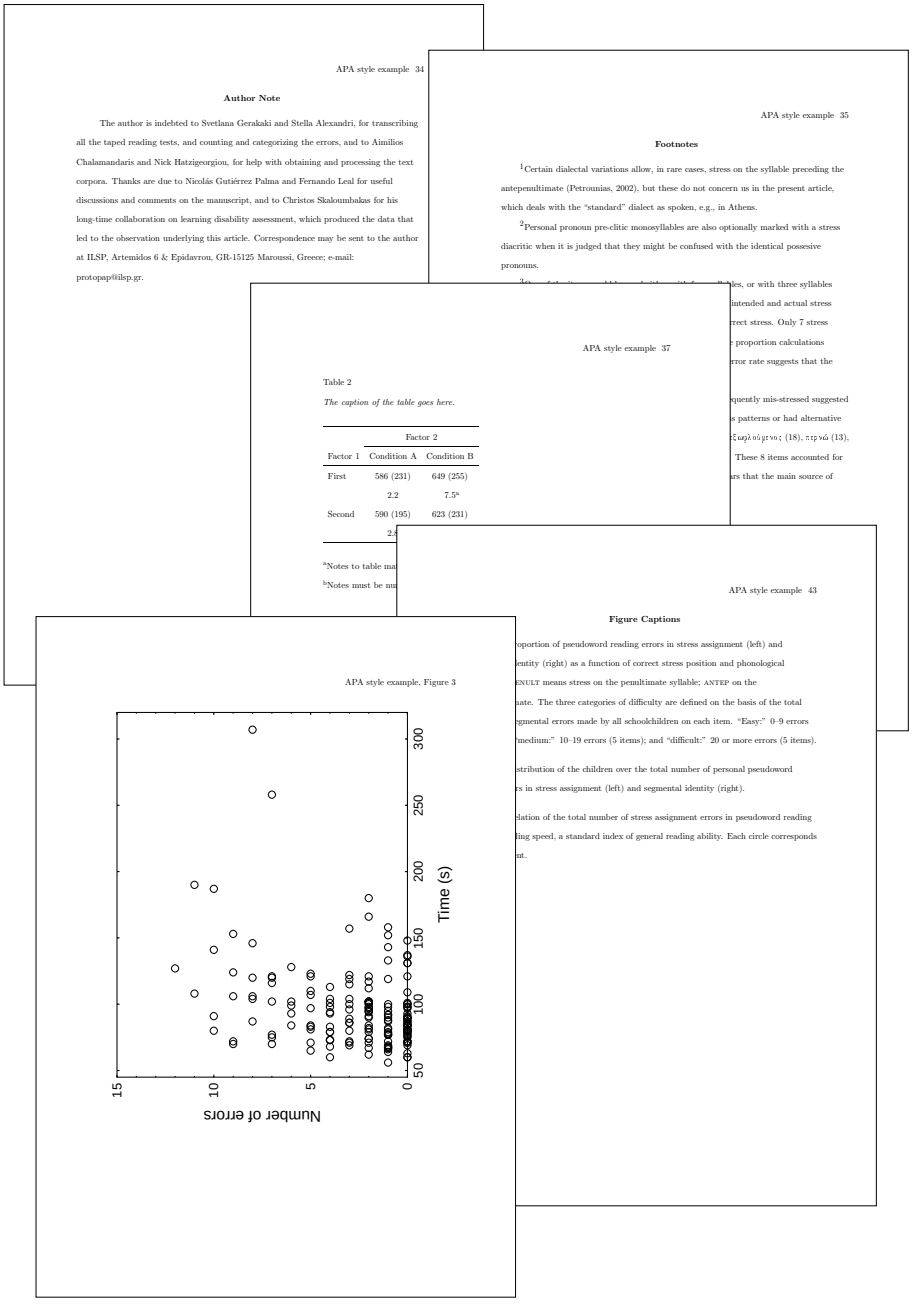


Figure 1: (Continued)

was solved most satisfactorily with the `apacite` package, which built somewhat on the earlier citation code of `theapa`, then extended and completed it while circumventing difficult encoding and memory limitation issues presented by the available LaTeX and BibTeX of that time. Because `apacite` is described by its author Erik Meijer elsewhere in this issue, and because it fully and completely addresses the citation and reference issues of APA style, these issues are not discussed further in the present article.

Thus the basic ingredients for creating the APA class were more or less present on the Internet of those times. I cannot overemphasize the role of both the Internet and the open TeX/LaTeX community, with the spirit of contribution and collaboration, in making useful larger-scale projects such as `apa.cls` feasible. Not only was I (and still am) insufficiently knowledgeable of both TeX and LaTeX to carry out such a project starting from scratch; but also the requirements in time investment would simply have been overwhelming for a project to tackle each and every peripheral issue arising in the battle with the monster called APA manual. Naturally, development of `apa.cls` also benefitted immeasurably from the contributions, comments, and extensive testing of many users and developers, as chronicled in the class changelog, which accompanies every release.

2.2 Conformance to requirements: problems, solutions, lessons, and side-effects

The manual requirements for lists, sectioning, footnote and float placement (at document end), and headers, were handled mostly by the aforementioned packages, even though some modifications to those packages were necessary in order to make their output conforming with the manual. These modifications highlighted once more the need for structured approaches to typesetting, because they presented novel instances of separating function (such as postponing a float till the end of the document) from appearance (such as selecting a typeface for displaying the float caption). Fitting caption display to the manual specifications was in fact sufficiently involved that no package could be parametrically modified at the time, leading to the inelegant but functional solution of copying from `latex.ltx` the code that implemented captions and modifying it at its deepest levels.

Setting the title page presented an interesting challenge, because it was desirable to retain the LaTeX concept of defining authors etc. in the preamble (and activating their appearance with a `\maketitle` command) while conforming to the manual rules and at the same time presenting a usable and helpful interface to the author that would help avoid mistakes. On the one hand, setting the authors in APA style is primarily dictated by how many affiliations are present, that is, in how many institutions the work was performed, because affiliations serve to group author names that are successive in the authorship sequence. On the other hand, author groups and affiliations must be provided to LaTeX separately from each other, not only because they constitute different

structural elements, but also (and more importantly) because they may have to be displayed differently or participate in different internal macros later on. The adopted solution, with a range of one up to six affiliations, as described in Section 3.1 below, fulfills these requirements and also provides consistency checking by verifying the number of declared author groups against the number of affiliations.

The rest of `apa.cls` constitutes primarily a specification of the appearance of a complete document, taking full control over the initial pages, using material declared in the preamble, as well as of the final pages, based on mechanisms defined in `endfloat` and `endnote`. In true LaTeX spirit, the class defines the entire document and uses packages to control local aspects of the text. The scope of `apa.cls` is thus very clearly defined to be the creation of manuscripts for publication in journals. This is important to stress because, as a document class, `apa.cls` is also cleary *inappropriate* for other types of documents, such as books or theses, even when such documents may share certain requirements with APA manuscripts, such as citation format, float placement, page design etc. As explained in the distribution and online documentation, `apa.cls` is most definitely *not* a document class for typesetting theses, books, or non-APA articles.

Given the basic structure and content of a manuscript that defines the necessary elements according to the manual guidelines, the complete separation of content from appearance naturally imposed by LaTeX permits alternative processing of the same source file with rather spectacular results. Figure 2 shows the same manuscript as presented in Figure 1 but processed with the `jou` ("journal" look) option of `apa.cls` instead of `man` ("manuscript" look). The appearance is based on journals published by APA as they are typeset by APA from the submitted manuscripts, because this is a format to which authors of technical psychological articles are accustomed and can easily follow its conventions. This format selection also allows authors to approximate the future look of their manuscripts (when they will be published articles) and make adjustments in tables, equations etc. (seeing, for example, whether they fit in a single column or not), taking into account the actual dimensions and constraints of published material.

An obvious additional advantage of this format is that it requires and uses precisely those structural elements that are present in the manuscript form, thus effectively obviating the need for any additional effort on the part of the author. The only exception to this is the publication-related information added by the journal publisher on the top of the first page of each article, for which special macros have been defined in `apa.cls`.

The implementation of the journal look has alleviated several problems presented by the manuscript look, such as end placement for floats and footnotes, while creating a few new ones, such as tight setting of several disparate elements on the first page and double-column balancing. To best approximate actual journal look, it was also necessary to use a Times-like typeface, with associated symbols. Similarly to the manuscript look, most of these problems were

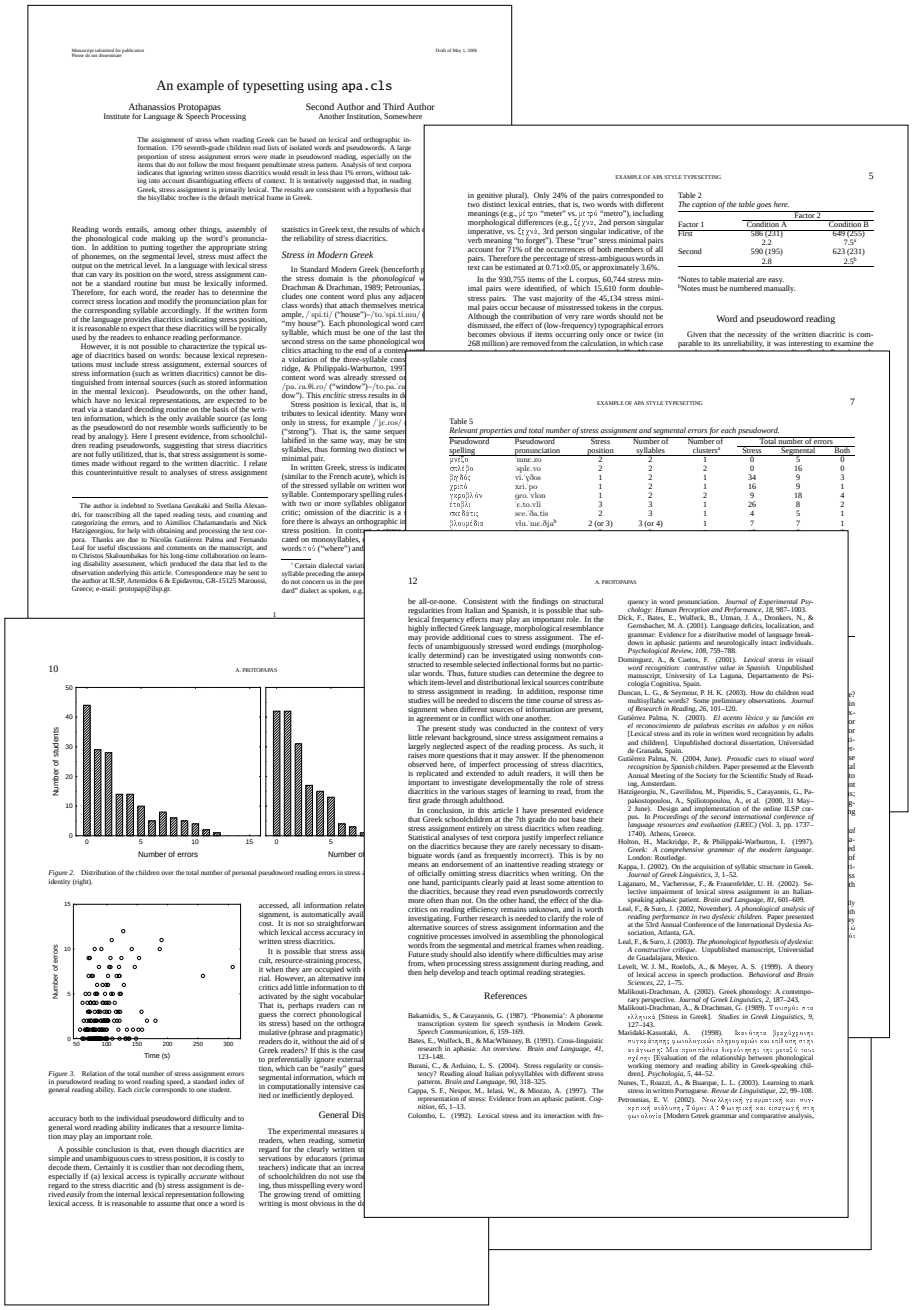


Figure 2: The same manuscript as in Figure 1 typeset in journal style with `apa.cls`.

addressed with pre-existing packages such as `pslatex` (more recently `txfonts`), `fancyhdr`, `flushend`, and `ftnright`. Because the appearance of this mode does not upset the order of manuscript elements it is in fact simpler to work with even though it has to meet stricter aesthetic expectations than the manuscript mode, in which page gaps and uneven lines are not to be avoided.

2.3 Summary: Alternative forms for the same content

In summary, `apa.cls` provides 3 alternative "looks" by typesetting the same document source according to different requirements. These are called "modes" and are passed as options to `apa.cls` in the `\documentclass` command (e.g., `\documentclass[man]{apa}`). They are:

Manuscript look (`man` mode) In `man` mode, `apa.cls` typesets a title page, abstract page, text body, then references, appendices, author notes, footnotes, tables, figure captions, and figures. All text is double spaced. The title page contains the (centered) title, author name, affiliation, short title, and the running head (automatically capitalized). A note may be inserted in place of each table and figure in the text. The title is printed as a heading at the beginning of the text. The "Author Notes" page contains the contents of `\acknowledgements`. A footnote page is produced after the bibliography ("References") if there is at least one footnote in the text.

Printed journal look (`jou` mode) In `jou` mode, `apa.cls` typesets the text in double-sided, double column format, with an appropriately centered title, author names, affiliations, note, and abstract. The contents of acknowledgements are printed as footnotes in a box at the lower left corner of the first page. Journal-like headers can also be produced on the first page. Dimensions and overall "look" are modeled after the *Journal of Experimental Psychology: Human Perception and Performance*.

Regular LaTeX look (`doc` mode) In `doc` mode, `apa.cls` typesets the text in single-sided, single-column format, with an appropriately centered title, author names, affiliations, note, and abstract on the first page. The contents of acknowledgements are printed at the bottom of the first page. Section headings and enumeration follow the APA guidelines. Floats (figures and tables) are inserted in the text where they are defined, as in regular LaTeX articles. The advantage of this mode over using a regular article class is that all the extra APA-related definitions are supported and no changes need to be made to the text in order to produce a LaTeX-looking article.

In the next section I present an overview of the instruments that were implemented in order to support all of the aforementioned functionality in the two most useful and important modes, that is, in man(uscript) and jou(rnal) mode.

3 Special macros and options

To use `apa.cls`, it is sufficient to declare it as the active document class at the top of the preamble using, for example, `\documentclass[man]{apa}`. The `man` option can be replaced by `jou` or `doc` for the other modes.

3.1 Title and headers

The appearance of the first page of both the manuscript and journal-style document is tightly constrained with respect both to the types and to the placement of information. In addition to the title and author names and affiliations, supplementary elements such as a short title and running head must be determined. `apa.cls` makes use of the standard `\title` command. However, since authors should be grouped by affiliation, the standard LaTeX `\author` command has been redefined and supplemented by the additional definitions of `\twoauthors`, `\threeauthors`, and so on up to `\sixauthors`, and the corresponding definitions of `\affiliation`, `\twoaffiliations`, `\threeaffiliations`, ... `\sixaffiliations`. The number of parameters passed to each of these commands is indicated in their names. The commands leading to the appearance shown in Figures 1 and 2 were:

```
\title{An example of typesetting using \texttt{apa.cls}}
\twoauthors{Athanassios Protopapas}%
           {Second Author and Third Author}
\twoaffiliations{Institute for Language \& Speech Processing}%
                {Another Institution, Somewhere}
```

`apa.cls` will issue an error message when the number of defined authors does not match the number of defined affiliations. Typesetting in the journal-style mode is done using first a `\tabular` environment to determine proper handling of widths (in case the column width is exceeded by an author-affiliation pair) and then setting the contents in a centering `\parbox` of the appropriate width, allowing the necessary vertical space.

Header information is specified using three special commands:

`\shorttitle` This is required by the APA manual, and is used in the header of the manuscript, along with the page number. It is only meant as a manuscript identifier and is not used in the published article. When this macro is undefined, `apa.cls` uses the full title.

`\rightheader` This is the running head for the published article, which must be specified by the author, according to the manual. It will appear on the right-hand pages of the published article (and in journal-look setting by `apa.cls`, naturally). In the manuscript mode, the contents of this macro appear as a "running head" defininition, as specified by the manual. When this macro is undefined, `apa.cls` substitutes the (short) title for the running head as well.

\leftheader The left-hand pages of the published APA-style articles carry the author names, typically with initials only for the first name. apa.cls will use the contents of \author if the left header is not defined.

For the first page of the journal-style look, special headers are defined, which correspond to the publication information displayed in two lines at the upper right and left corners of the APA journals (journal name, volume/issue information, copyright, and archiving information). The four corresponding macros \journal, \volume, \ccoppy, and \copnum affect typesetting in jou and doc mode (as shown in Figure 2) and are ignored in man mode.

3.2 Additional preamble definitions

The first page of published articles, APA-style or otherwise, typically includes additional special material, such as an abstract and acknowledgements (in APA-style parlance termed "author notes" and including information such as current full addresses of the authors, previous presentations of the material etc.). In manuscript submission, the contents of both the abstract and the acknowledgements must appear in separate pages, specifically the abstract on the 2nd page of the manuscript and the acknowledgements in a page titled "Author notes" following the article text and bibliographic references. apa.cls handles these requirements through two macros, namely \abstract and \acknowledgements, each admitting a single long-text argument.

The contents of the argument to \abstract are displayed as part of the title setting in both modes (i.e., within \maketitle). In manuscript mode, a new page is started and the heading "Abstract" is displayed before the abstract text, whereas in journal mode the abstract text is displayed in a paragraph of the appropriate width (less than \textwidth) under the title and author-affiliation information.

The contents of the argument to \acknowledgements are displayed in a separate page after the document text in manuscript mode, in an \AtEndDocument construction (also used to display footnotes as endnotes), under the heading "Author notes." In journal mode, this text is displayed in a figure environment set to the bottom of the first page in \footnotesize, containing a \parbox of \columnwidth below a separating horizontal line (made with \vrule). Using a floating environment allows the article text to float around this material, which must be fixed to the lower-left corner of the first page regardless of its size.

An additional \note macro, with a single argument, is provided for adding author notes (in the colloquial, not APA, sense) to the document first pages. This macro is often useful for indicating on the title page the author contact information, as many journal editors require (in violation of the APA manual rules for setting the manuscript title page), and as shown also in Figure 1 above. The contents of this macro argument are displayed above the abstract in journal mode, where it is unofficial yet customary to add indications such as the draft date or status of a manucript disseminated prior to publication.

3.3 Sectioning

APA-style sections are not numbered. This creates awkward situations when one needs to refer back (or ahead) to a section, because the full heading must be used, for example here we might want to direct the reader to the previous section by referring to Section *Additional preamble definitions*. Apart from this issue, sectioning can be complicated because the appearance of each heading level depends on the number of levels employed in a given manuscript. That is, the top-level, second-level etc. heading style may differ in appearance depending on whether there are two or four levels in all. The manual defines specifications for up to five levels of headings, and `apa.cls` incorporates all of these specifications, largely reusing code from `theapa.sty` by Young U. Ryu, with minor formatting modifications necessary to conform to the more recent manual editions. Thus, `apa.cls` offers the five standard LaTeX sectioning commands: `\section`, `\subsection`, `\subsubsection`, `\paragraph`, and `\subparagraph`.

The number of levels used in a document is specified with the additional `\headinglevels` macro, provided by `apa.cls`, which takes exactly one argument, specifically one of the words "one," "two," "three," "four," and "five," and adjusts the appearance of each heading level accordingly. Because most articles use two or three levels of headings, and there are no appearance differences between these two cases, `apa.cls` assumes three levels by default, and this need not be changed in most cases. This way authors typically need only concern themselves with the standard sectioning commands `\section`, `\subsection`, and perhaps `\subsubsection`, and not with the number of heading levels.

Needless to say, `apa.cls` adjusts the formatting of section headings differently depending on the processing mode, since manuscript specifications are different from the look of journal publications.

3.4 Enumeration and seriation

The manual defines three cases of what in LaTeX we would call "list environments," that is, sets of items that enumerate or itemize cases. In fact, two of these APA-specific cases are conceptually indentical to the standard LaTeX `enumerate` and `itemize` environments, simply necessitating different horizontal and vertical space settings, that is, margins and (no) space between items.

The third case is different and is actually used most often in APA-style manuscripts. Its characteristic is that it does not begin each item on a new line but uses alphabetical in-parentheses enumeration to display the list "inline." As a relevant example, we could state here that `apa.cls` offers three list environments: (a) the `enumerate` environment, which uses roman numerals, regular paragraph identation and margins, and starts a new line with each item; (b) the `itemize` environement, which uses bullet symbols instead of numerals and is otherwise identical to `enumerate`; and (c) the `seriate` environment, which produces in-line lists with lowercase letter enumeration such as this very exam-

ple. The first two are redefinitions of the standard list environments whereas the third simply keeps track of an `\alph` counter. All three APA list environments were implemented by Young U. Ryu in `theapa.sty` and are incorporated unmodified in `apa.cls`.

3.5 Figures and tables

3.5.1 About floats in manuscripts

Figures and tables are floating bodies and, in published articles, they are set at a page corner near where they are first mentioned, without interrupting the flow of the text. In LaTeX this behavior is taken for granted, but most authors of psychological manuscripts have not in general used LaTeX to produce their articles. APA seems to have heard of computers only relatively recently, since up to the 4th (1994) edition of the manual everything appeared geared toward typewriters. The specifications of the 5th (2001) edition are still far from incorporating the change in workstyle dictated by the invasion of computers into office tasks and consequent significant facilitation and improvements.

This is to say that the manual requires tables to be set after the main body of the manuscript (and references), each with its own caption on a separate page, while figures follow, at the very end of the manuscript, after a separate page containing the figure captions. All of this, of course, is a relic of the days in which diagrams were made by hand on special paper, or special material was photographed etc. Until such time arrives that APA is made aware of computer files, printers, scanners, and so on (not to mention structured document specifications and XML), we are all still obliged to postpone display of tables and figures till the end of the manuscript, a demanding task that is fortunately very completely taken care of by the `endfloat` package.

`apa.cls` loads `endfloat`, in manuscript mode, and modifies it somewhat in order to match the requirements of the manual as far as appearances and captions are concerned. For tables, which are processed before figures, a new page is started with each table. The author is responsible for defining the table caption, within the `table` environment, *before* the `tabular` environment in which the main table material is set. Additional rules in the manual dictate the proper use of horizontal and (no) vertical separators.

`apa.cls` makes two provisions for the journal style only: the `tabular` environment automatically stretches to occupy the full column width (or double column width, inside of a `table*` environment), and the new macro `\thickline` produces a double-height horizontal separator, useful for the top of the table according to the APA journal look. None of these effects are present in the manuscript mode, of course.

3.5.2 Notes in tables

Tables in APA manuscripts frequently need special use of notes, which are restricted to the current table and are displayed immediately below the bot-

tom separator. There are very substantial difficulties in defining table-footnote macros that will automatically handle numbering, however such an exercise would often be of little use because table notes frequently make repeated use of the same marks (letters, for actual notes, and one or more asterisks, for levels of statistical significance). Thus what is really needed in a table is not a way to automatically number notes, but a way to insert marks that will not mess up column alignment, and a complementary macro to repeat those marks along with the corresponding definition or explanation below the table.

`apa.cls` provides `\tabfnm`, which takes a single argument, for adding note marks, as superscripts, in the tabular material, and `\tabfnt`, which takes two arguments, for providing the marks along with the corresponding text after the table. It is up to the author to match marks between the two macros, but this allows the flexibility of using each mark as often as needed.

In summary, here is a proper, complete definition of a table for an APA-style manuscript in `apa.cls`, illustrating the use of the aforementioned commands:

```
\begin{table}[tbp]
\caption{The caption of the table goes here.}
\label{tab:tab1}
\begin{tabular}{lcc}\thickline
          & \multicolumn{2}{c}{Factor 2} \\ \cline{2-3}
Factor 1  & Condition~A  & Condition~B   \\ \hline
First     & 586 (231)    & 649 (255)     \\
          & 2.2          & 7.5\tabfnm{a} \\
Second    & 590 (195)    & 623 (231)     \\
          & 2.8          & 2.5\tabfnm{b} \\ \hline
\end{tabular}
\tabfnt{a}{Notes to table material are easy.}
\tabfnt{b}{Notes must be numbered manually.}
\end{table}
```

Note that `\caption` precedes the tabular environment and that `\label` immediately follows the caption. The results of this definition are seen in Figures 1 and 2, in displayed document pages 37 and 5, respectively.

3.5.3 Large floats

Tables too wide to display in "portrait" orientation can be set sideways using the `rotating` package and the `\sidewaystable` environment (provided that `endfloat` is propertly configured to recognize this environment; the needed configuration file is provided with `apa.cls`).

Tables too long to fit on a single manuscript page pose special problems for `apa.cls`. Note that we do not refer here to tables too long to fit on a single *journal* page, for that would pose manual typesetting requirements well outside the scope of `apa.cls`. It is not too infrequent, however, for a table to fit in a journal page but not in a manuscript page, because of the wider

margins and line spacing, larger font, and space devoted to the caption. The standard usage of `endfloat` is not sufficient to handle this case. Fortunately, two solutions exist: a "proper" one, with a LaTeX package, and an ad-hoc one, with no special provisions.

Unfortunately, both of these solutions make the document unusable with the journal mode, so they are more difficult to maintain and defeat an important purpose of `apa.cls`. The "proper" solution is to use the `longtable` package, with its special macros. Due to precedence requirements with `endfloat`, the package should be declared as an option to `apa.cls` and may not be loaded directly. A special `endfloat` configuration file is also necessary (and supplied with `apa.cls`). For authors familiar with `longtable`, this allows them to produce the long tables they need. For others, a less clean option may be taken, as shown in the following example:

```
\begin{table}[tbp]
\caption{This is the regular caption of our long table.}
\begin{tabular}{lcc}\thickline
Factor  & Condition~A  & Condition~B   \\ \hline
First   & 586 (231)    & 649 (255)     \\
Second  & 590 (195)    & 623 (231)     \\

⋮

\end{tabular}
\end{table}
\begin{table}[tbp]
\addtocounter{table}{-1}
\caption{(Continued)}
\begin{tabular}{lcc}\thickline
Factor    & Condition~A  & Condition~B   \\ \hline
Thirtieth & 865 (312)    & 496 (552)     \\

⋮

\end{tabular}
\end{table}
```

That is, one simply interrupts a long table, in the appropriate point of the tabular environment, with a table end followed by a table beginning. This presents two tables to `endfloat`, so it can be processed correctly, while making the result appear as a single table, with continuation, to the editor who will receive the manuscript, because of the reduction of the table counter. The disadvantage of this option is that the author must determine manually where to split the table; the advantage is that no special macros, packages, or configurations are necessary.

The main reason that none of the two solutions is compatible with `jou` mode is that (table) environment beginnings and endings may not be inside conditional compilation macros and can be neither renamed nor moved because that would make them invisible to `endfloat`.

3.5.4 Figure sizing

The situation of figures is simpler, assuming that the manuscript author has already produced the needed diagram or other pictorial material in Encapsulated Postscript (`.eps`) format (or in Portable Document Format, `.pdf`, for use with pdfLaTeX). In contrast to tables, figure captions are placed after the figure material, consistent with their appearance *below* the diagrams in the published journal articles.

Figures present an interesting case of difference between manuscript and journal modes: diagrams are typically set flush with the column margins in published articles, while it is desirable to enlarge them to occupy as much of a page as possible when submitting a manuscript, in order to improve printed quality via shrinking in photographic reproduction. For vector graphics images the situation is simple, because there is no loss of display quality when enlarging to fit the available page. Therefore `apa.cls` offers the `\fitfigure` macro, which makes a call to `\includegraphics` of the `graphicx` package, which must be loaded in the preamble by the author. This sets the figure size to the available column width, in journal mode, or to the page width, automatically rotating when necessary, to best match the aspect ratio of the figure, in manuscript mode (see displayed pages "Figure 3" and 10 of Figures 1 and 2).

A figure definition example is:

```
\begin{figure}[tp]
\fitfigure{filename}
\caption{This is the caption of the figure.}
\label{fig:name}
\end{figure}
```

Here, using simply "filename" (instead of "filename.eps") as the argument to `\fitfigure` ensures that the `.eps` file will be loaded when processed by LaTeX and `dvips`, whereas the `.pdf` file will be loaded when procesed by pdfLaTeX (assuming that the author has made both format versions of the image available in a searchable directory/folder).

In contrast to a vector image such as a diagram, if the author wishes to include a bitmap image with the submitted manuscript, this would not be improved by stretching; on the contrary, reproduction quality might be adversely affected with any modification of the "natural" resolution of the bitmap. Thus `apa.cls` offers an additional macro named `\fitbitmap`, which also takes the graphics file name as a single argument. This macro fits the image to the column width in journal mode but does not affect the image in any way in manuscript mode.

3.5.5 Captions

Handling of captions in `apa.cls` is cumbersome and potentially incompatible with future versions of LaTeX or `endfloat`. The need to affect both the content and appearance of the caption heading (e.g., the word "Table" or "Figure"), along with the requirements for typesetting the caption text, have necessitated modifications at the deepest level of the caption definitions, which are not normally available in LaTeX. Thus when `apa.cls` was written, the definitions of `\caption`, `\@caption`, `\@@caption`, `\@@@caption`, and `\@makecaption` were copied from the then-current `latex.ltx` and appropriately modified to produce the desired effects. This is an unfortunate situation, which may or may not be ameliorated by using one of the now available caption handling packages, such as `ccaption` or `caption2`, in future versions of `apa.cls`.

3.6 Internationalization

Although the manual is meant for English-language articles only, it is not uncommon for psychological journals in non-English-speaking countries to require APA-style manuscripts, with minor modifications. Thus there is a definite potential for international use of `apa.cls` within its strict scope (of producing APA-style manusripts).

In general, language matters such as hyphenation and certain document-related words, are handled by the now standard `babel` package. Because bibliographies are handled by the `apacite` package (see accompanying article by Erik Meijer in this issue), internationalization issues related in particular to citations and references are already fully taken care of. Thus there is no need for `apa.cls` to mess with definitions of "and," "editor," and so on. There are, however, a few strings that are produced by `apa.cls` itself, particularly in the manuscript mode, including "Abstract," "Author Note," "Figure Captions," "Footnotes," (as page headers), and "Running head" (on the first page). Of these, "Abstract" is handled by `babel`, but the rest are included as text to be displayed by `apa.cls`.

In older versions, due to the nationality (and consequent personal needs) of the author, `apa.cls` checked at the beginning of the document if Greek was the active (selected) language, and modified these strings if that was the case. With the recent improvement of `apacite`, including handling of certain potential incompatibilities with `babel` as well as an extensible internationalization scheme, it was also time for `apa.cls` to follow suit and define a set of external files for deriving language-specific strings as needed. This has been implemented in 2006 as of version 1.3.2 of `apa.cls`, using the "`.apa`" file extension. For example, `dutch.apa` contains the four Dutch definitions, `greek.apa` the Greek definitions, and so on. Since there is no internal language-handling code, any user may add definitions for a language by simply creating a text file using the installed babel name for the new language.

3.7 Conditional compilation

`apa.cls` was designed so that it would not be necessary for the author to be concerned with the final output format of the manuscript. Therefore every effort has been made for things to be taken care of automatically as much as possible. However, unforeseeable circumstances are bound to occur. In addition, (non-APA) journal editors may impose additional requirements to authors, not entirely in line with the manual yet not too far deviating from it. A relatively frequent example is the requirement for putting author contact information on the title page, a practice not dictated by the manual. For this and other cases in which the author is forced to produce content differences between manuscript look and journal look, a set of conditional compilation macros is provided, in which evaluation depends on the mode currently processed. Specifically,

`\ifapamode{text1}{text2}{text3}` produces "text1" in man mode, "text2" in jou mode, and "text3" in doc mode;

`\ifapamodeman{man-mode text}{other text}` makes a distinction between man mode and the others; and similarly

`\ifapamodejou{jou-mode text}{other text}` and

`\ifapamodedoc{doc-mode text}{other text}` provide for mode-specific content for each of the other modes.

These macros provide much useful functionality; however they cannot solve all the problems that may arise, such as for example different requirements for floating environments, as mentioned above, because environments may not be placed in arguments, and furthermore `endfloat` would not work properly with the float delimiters hidden inside conditional compilation macros.

3.8 Fonts

The manual states a preference for serif typefaces and expressly recommends "12-pt Times Roman" and "12-pt Courier" (p. 285). Presumably they would be happy with Computer Modern fonts as well. For authors who wish to accentuate the perversity of typewriter-oriented instructions in the year 2005, `apa.cls` provides the `tt` option, which gives a quaint typewritten appearance to the manuscript (and is duly ignored in journal mode, naturally). Other than this, authors need not and should not concern themselves with fonts, except maybe to select a readable sans font for their figure labels and axis legends in their diagram-drawing software.

For the journal look, `apa.cls` tries to approach the *Journal of Experimental Psychology* as much as possible, searching for `txfonts`, `pslatex`, or `times/mathptm`, in this order, and loading the first of these that is found. For the author who likes the layout but does not wish to produce a document that looks too much like published when it is not, the option `notimes` to `apa.cls` (taken into account in `jou` mode only) reverts to Computer Modern.

3.9 Appendices

Appendices are frequently used in psychological articles, for lists of stimuli, supplementary tabular material, secondary explanations, or mathematical derivations etc. In APA style, appendices are numbered (using capital Roman letters) if there two or more of them; but a single appendix is simply "Appendix" and not "Apendix A." In `apa.cls`, this was achieved based on code contributed by Michael Erickson, while not departing from standard LaTeX usage. Specifically, after the bibliography, the manuscript author declares simply

```
\appendix
```

and then begins each appendix (even if there is only one) with a `\section` command, including the appropriate appendix heading. For example,

```
\section{List of words}
\label{app:stim}
```

will produce an appendix titled "List of words" (shown in Figures 1 and 2). Note that the `\appendix` command does not create any appendices but only switches to the appropriate heading and numbering routines that will be triggered by sectioning. Because appendices are numbered, it is possible to refer to them using the internal LaTeX reference system. So you can insert an optional `\label` command right after each appendix `\section`, as in the above example, and then refer to the corresponding appendix with a `\ref` macro; for the example the reference would be

```
see Appendix~\ref{app:stim}
```

4 Maintenance and support

As mentioned before, `apa.cls` is both a personal project and the product of a communal effort. Even though it originated by the need and desire of a single person for an easier approach to APA-style manuscripts, in the LaTeX spirit, and was largely implemented single-handedly, four major types of contributions must be acknowledged: The developers of packages and solutions that are used for specific effects, several contributors to `comp.text.tex` for help and advice whenever asked, developers who sent suggestions and sometimes even code to solve particular problems or add functionality to `apa.cls`, and many users who tested it extensively and sent detailed problem reports, helping to identify and eliminate problems.

Unfortunately, due to lack of time, there is no "official" support or regular maintenance of `apa.cls`. However, at this point most issues have been largely resolved (or circumvented) and it has become very rare to receive a report of a problem, even including problems not caused by `apa.cls`. Ideally, updates and revisions would be made to reflect the progress made with LaTeX and the standard packages and distributions. It is possible that improvements in

caption and float packages, in particular, may have rendered obsolete certain workarounds and modifications made by `apa.cls`. If this is the case, it would be best for `apa.cls` to simply use the new version of these packages unmodified, passing appropriate parameters as needed.

Removing kludges and updating external references would help `apa.cls` remain current in an ever-changing computing environment, or at least remain usable in the foreseeable future despite major upgrades in operating systems, distributions, and even with LaTeX itself (Omega, Lambda, and the LaTeX3 project being the most imminent "threats"). Unfortunately, time does not permit such drastic measures or even regular active maintenance. For this reason, if anyone is interested in taking over maintenance of `apa.cls` and helping it migrate fully to the current and future contingencies, I would be delighted and greatly relieved to "hand over" the honor.

In the meantime, `apa.cls` can be found at `CTAN:/tex-archive/macros/latex/contrib/other/apa/` and through the TeX Catalogue. The current version as of this writing is 1.3.2. Instructions for using `apa.cls` can be found in the distribution, and on the dedicated web page maintained by the author at `http://www.ilsp.gr/homepages/protopapas/apacls.html`.

Acknowledgements

Ioannis Dimakos of the Greek TeX Users Group ($\epsilon\phi\tau$) and Erik Meijer, the creator of `apacite`, have been instrumental in the conception and realization of the present manuscript. Erik has also tremendously helped the progress of `apa.cls` by agressively maintaining and improving `apacite` such that concomintant revisions of `apa.cls` were necessary; and by testing and even contributing code for `apa.cls` to solve issues well outside the domain of my experience.