langsci-affiliations

Felix Kopecky*

Version 1.3 – 27th February 2024

1 User guide

This package provides a command `\ResolveAffiliations`, which collects author–affiliation pairs and outputs them according to the user configuration. It is aimed at class authors, i.e. maintainers of document templates in publishing houses, universities, etc.

`\ResolveAffiliations` `\ResolveAffiliations [⟨options⟩] {⟨pairs of authors and affiliations⟩}`

Takes the `{⟨pairs of authors and affiliations⟩}`, orders them internally and outputs them according to the `[⟨options⟩]`.

`{⟨Pairs of authors and affiliations⟩}` is a list of authors and affiliations, separated by a customisable string. The defaults for the separators are `and` for authors and `;` for affiliations. The conventional author separator `\and` is automatically converted to the chosen author separator. Affiliations are given within `\affiliation` within the `{⟨pairs⟩}` argument. This command is not defined by this package and possibly existing definitions are left unchanged.

For example:

```
\ResolveAffiliations{
  A. U. Thor\affiliation{University of the Moon; University of Mars}
  and B. U. Thor\affiliation{University of Mars}
}
```

results in:

A. U. Thor[a,b] & B. U. Thor[b]

[a]University of the Moon [b]University of Mars

The output can be customised using the `[⟨options⟩]`. They are described below.

`\SetupAffiliations` `\SetupAffiliations {⟨options⟩}`

Options can be set either globally or locally. With `\SetupAffiliations{⟨options⟩}`, they apply globally. If they are set with `\ResolveAffiliations[⟨options⟩]`, they apply locally.

*mailto:felix.kopecky@langsci-press.org. Please submit bug reports and feature requests to https://github.com/langsci/langsci-affiliations/issues.

`mark style` = ⟨*style*⟩ (initially `alphabetic`)
Controls which markers should be used in the indexes of affiliations. Can be a either of {`alphabetic`, `numeric`, `circled`, `none`}.

`output affiliation` = ⟨*boolean*⟩ (initially `true`)
Affiliations are output if true, otherwise not.

`orcid placement` = ⟨*choice*⟩ (initially `none`)
Decide whether and where to place ORCIDs around author names. Valid choices are {`none`, `before`, `after`}.

`output in groups` = ⟨*boolean*⟩ (initially `true`)
If `true`, authors and affiliations are output in the same line. When `false` each author and affiliation gets its own line. Only available if `output affiliation=true`.

`output authors font` = ⟨*font commands*⟩ (initially `\Large`)
Stores the font settings for the ouput of authors.

`output affiliation font` = ⟨*font commands*⟩ (initially `\normalsize`)
Stores the font settings for outputting affiliations.

`output authors paragraph format` = ⟨*layout settings*⟩ (initially `\raggedright`)
Stores the paragraph settings for the author block. These settings are only applied if `output in groups=true`.

`output affiliation paragraph format` = ⟨*layout settings*⟩ (initially `\raggedright`)
Stores the paragraph settings for the affiliation block. These settings are only applied if `output in groups=true`.

Output separators between authors and affiliations are customisable as well:

`separator between two` = ⟨*tokens*⟩ (initially `~&~`)
If there are only two authors, use these ⟨*tokens*⟩ to separate them.

`separator between multiple` = ⟨*tokens*⟩ (initially `,~`)
If there are more than two authors, use these ⟨*tokens*⟩ to separate every pair except the last one.

`separator between final two` = ⟨*tokens*⟩ (initially `~&~`)
Use these ⟨*tokens*⟩ to separate the last pair of authors if ther are more than two.

`separator between indices` = ⟨*tokens*⟩ (initially `,`)
Use these to separate affiliation indices after each author.

`separator between affiliations` = ⟨*tokens*⟩ (initially ␣)
Separates the affiliations in the affiliation line.

The way the input is digested can be customised with these two settings:

`input names separator` = ⟨*tokens*⟩ (initially `~and~`)
Separates the author names in the input.

`input affiliation separator` = ⟨*tokens*⟩ (intially `;`)
Separates the affiliations in the input, within dummy command `\affiliation`.

\CountAuthorsFromAffiliations

New: 2021-12-06

\CountAuthorsFromAffiliations [⟨*options*⟩] {⟨*pairs of authors and affiliations*⟩}

A document command to count the numbers of authors given in a list. Useful for conditional behaviour of document classes based on the numbers of authors. It takes the same optional arguments as \ResolveAffiliations. For example, a custom author separator is recognised by this command.

The result is stored in the global integer variable \g__affiliations_num_authors_int.

\LinkToORCIDinAffiliations

New: 2022-09-27

\LinkToORCIDinAffiliations {⟨*orcid*⟩}

This document command is intended as a user interface to customise the way ORCIDs are output. For example, it can be set to forward the input ORCIDs to \orcidlink from the orcidlink package:

```
\RenewDocumentCommand{\LinkToORCIDinAffiliations}{ +m }
  {%
    \,\orcidlink{#1}%
  }
```

2 Implementation

```
⟨*package⟩
⟨@@=affiliations⟩
\RequirePackage{xparse}
\ProvidesExplPackage {langsci-affiliations} {2024-02-27} {1.3}
{A LaTeX3 package to collect and order authors and affiliations}
```

\ResolveAffiliations The top-level document command. It is grouped to keep assignments local.

```
\NewDocumentCommand{\ResolveAffiliations}{ O{} +m }
  {%
    \group_begin:
    \keys_set:nn { affiliations } { #1 }%
    \exp_args:No \affiliations_resolve:n { #2 }%
    \group_end:
  }
```

(*End of definition for* \ResolveAffiliations*. This function is documented on page* 1.)

\LinkToORCIDinAffiliations The action taken to link to an ORCID. Designed to be overwritten by the user.

```
\ProvideDocumentCommand{\LinkToORCIDinAffiliations}{ +m }{ #1 }
```

(*End of definition for* \LinkToORCIDinAffiliations*. This function is documented on page* 3.)

\CountAuthorsFromAffiliations Count authors and leave the result in the global integer variable \g__affiliations_num_authors_int.

```
\NewExpandableDocumentCommand{\CountAuthorsFromAffiliations}{ O{} +m }
  {%
    \group_begin:
    \keys_set:nn { affiliations } { #1 }%
```

```
    \exp_args:No \affiliations_count_authors:n { #2 }%
    \group_end:
  }
```

(*End of definition for* `\CountAuthorsFromAffiliations`. *This function is documented on page* 3.)

`\SetupAffiliations` A command to define options.

```
\NewDocumentCommand{\SetupAffiliations}{ m }
  {%
    \keys_set:nn { affiliations } { #1 }
  }
```

(*End of definition for* `\SetupAffiliations`. *This function is documented on page* 1.)

```
\keys_define:nn { affiliations }
  {
    mark~style .tl_set:N
        = \l__affiliations_style_tl,
    mark~style .initial:n
        = { alphabetic },
    output~affiliation .bool_set:N
        = \l__affiliations_output_affiliation_bool,
    output~affiliation .initial:n
        = { true },
    orcid~placement .tl_set:N
        = \l__affiliations_orcid_place_tl,
    orcid~placement .initial:n
        = { none },
    output~in~groups .bool_set:N
        = \l__affiliations_output_grouped_bool,
    output~in~groups .initial:n
        = { true },
    separator~between~two .tl_set:N
        = \l__affiliations_separator_between_two_tl,
    separator~between~two .initial:n
        = {~\&~},
    separator~between~multiple .tl_set:N
        = \l__affiliations_separator_between_mult_tl,
    separator~between~multiple .initial:n
        = {,~},
    separator~between~final~two .tl_set:N
        = \l__affiliations_separator_between_last_two_tl,
    separator~between~final~two .initial:n
        = {~\&~},
    separator~between~indices .tl_set:N
        = \l__affiliations_indices_separator_tl,
    separator~between~indices .initial:n
        = {,},
    separator~between~affiliations .tl_set:N
        = \l__affiliations_afil_separator_tl,
    separator~between~affiliations .initial:n
        = {~},
    output~authors~paragraph~format .cs_set:Np
        = \__affiliations_output_authors_paragraph_format:,
    output~authors~paragraph~format .initial:n
```

```
      = {\raggedright},
  output~authors~font .cs_set:Np
      = \__affiliations_output_authors_font:,
  output~authors~font .initial:n
      = {\Large},
  output~affiliation~font .cs_set:Np
      = \__affiliations_output_affiliation_font:,
  output~affiliation~font .initial:n
      = {\normalsize},
  output~affiliation~paragraph~format .cs_set:Np
      = \__affiliations_output_affiliation_paragraph_format:,
  output~affiliation~paragraph~format .initial:n
      = {\raggedright},
  input~names~separator .tl_set:N
      = \l__affiliations_input_names_sep_tl,
  input~names~separator .initial:n
      = {~and~},
  input~affiliation~separator .tl_set:N
      = \l_affiliations_input_afil_sep_tl,
  input~affiliation~separator .initial:n
      = {;}
}
```

\prop_put:Nxx
\prop_put:Nnx
\seq_set_split:Nvn
\l__affiliations_tmpa_clist
\l__affiliations_tmpa_int
\g__affiliations_num_authors_int
\l__affiliations_affiliations_seq
\l__affiliations_authors_seq
\l__affiliations_names_seq
\l__affiliations_tmpa_seq
\l__affiliations_tmpb_seq
\l__affiliations_tmpa_tl
\l__affiliations_tmpb_tl
\l__affiliations_output_prop
\l__affiliations_affiliations_prop

Variants and variables

```

\cs_generate_variant:Nn \prop_put:Nnn { Nxx }
\cs_generate_variant:Nn \prop_put:Nnn { Nnx }
\cs_generate_variant:Nn \seq_set_split:Nnn { NVV }
\cs_generate_variant:Nn \seq_set_split:Nnn { NVn }
\cs_generate_variant:Nn \tl_replace_all:Nnn { NnV }
\clist_new:N \l__affiliations_tmpa_clist
\int_new:N \l__affiliations_tmpa_int
\int_new:N \g__affiliations_num_authors_int
\seq_new:N \l__affiliations_affiliations_seq
\seq_new:N \l__affiliations_authors_seq
\seq_new:N \l__affiliations_names_seq
\seq_new:N \l__affiliations_tmpa_seq
\seq_new:N \l__affiliations_tmpb_seq
\seq_new:N \l__affiliations_tmp_affil_seq
\seq_new:N \l__affiliations_tmp_orcid_seq
\tl_new:N \l__affiliations_tmpa_tl
\tl_new:N \l__affiliations_tmpb_tl
\tl_new:N \l__affiliations_tmpc_tl
\prop_new:N \l__affiliations_tmpa_prop
\prop_new:N \l__affiliations_output_prop
\prop_new:N \l__affiliations_affiliations_prop
\prop_new:N \l__affiliations_orcids_prop
```

(*End of definition for* `\prop_put:Nxx` *and others.*)

\l__affiliations_icons_prop

The data for the `circled` mark style. Since this uses the `\char`, it is only available in XeLaTeX.

```
\prop_const_from_keyval:Nn \l__affiliations_icons_prop
  {
```

```
    0 = \char"2460,  1 = \char"2461,  2 = \char"2462,  3 = \char"2463,
    4 = \char"2464,  5 = \char"2465,  6 = \char"2466,  7 = \char"2467,
    8 = \char"2468,  9 = \char"2469, 10 = \char"246A, 11 = \char"246B,
   12 = \char"246C, 13 = \char"246D, 14 = \char"246E, 15 = \char"246F,
   16 = \char"2470, 17 = \char"2471, 18 = \char"2472, 19 = \char"2473
}
```

(*End of definition for* `\l__affiliations_icons_prop`.)

`\__affiliations_resolve_affiliations:` A helper macro to order affiliations. Is called by `\affiliations_resolve:n`.

```
\cs_new:Npn \__affiliations_resolve_affiliations: #1#2
  {
    \clist_clear:N \l__affiliations_tmpa_clist
    \tl_if_empty:nTF {#2}
      {
        \prop_put:Nnn \l__affiliations_output_prop {#1} {}
      }
      {
        \seq_set_split:NVn \l__affiliations_tmpa_seq
                           \l_affiliations_input_afil_sep_tl
                           { #2 }
        \seq_map_inline:Nn \l__affiliations_tmpa_seq
          {
            \prop_get:NnNTF \l__affiliations_affiliations_prop
                            {##1}
                            \l__affiliations_tmpa_tl
              {
                \clist_put_right:NV \l__affiliations_tmpa_clist
                                    \l__affiliations_tmpa_tl
              }
              {
                %Not yet present
                \clist_put_right:Nx \l__affiliations_tmpa_clist
                  {
                    \prop_count:N \l__affiliations_affiliations_prop
                  }
                \prop_put:Nnx \l__affiliations_affiliations_prop {##1}
                  { \prop_count:N \l__affiliations_affiliations_prop }
              }
          }
        \prop_put:NnV \l__affiliations_output_prop
                      {#1}
                      \l__affiliations_tmpa_clist
      }
  }
```

(*End of definition for* `\__affiliations_resolve_affiliations:`.)

`\__affiliations_output_affiliations:` A helper macro that outputs the list of affiliations, usually below the list of authors.

```
\cs_new:Nn \__affiliations_output_affiliations:
  {
    \prop_map_inline:Nn \l__affiliations_affiliations_prop
      {
        \int_set:Nn \l__affiliations_tmpa_int { ##2 }
        \str_case_e:nn { \l__affiliations_style_tl }
```

```
          {
            {alphabetic}
              {
                \textsuperscript{\int_to_alph:n{ \int_eval:n
                  {  \l__affiliations_tmpa_int + 1  }
                } }
              }
            {numeric}
              { \textsuperscript{\int_eval:n {\l__affiliations_tmpa_int + 1} } }
            {circled}
              {
                \prop_item:Nn \l__affiliations_icons_prop
                              {\l__affiliations_tmpa_int}
              }
            {none} { }
          }
        \tl_rescan:nn {} {##1}
        \int_compare:nNnT
          { \int_eval:n {\l__affiliations_tmpa_int + 1} }
          <
          { \prop_count:N \l__affiliations_affiliations_prop }
          { \tl_use:N \l__affiliations_afil_separator_tl }
      }
  }
```

(*End of definition for* `\__affiliations_output_affiliations:`.)

`\__affiliations_return_afil_text:n` A helper macro that returns the affiliation marks.

```
\cs_new:Npn \__affiliations_return_afil_text:n #1
  {
    \int_set:Nn \l__affiliations_tmpa_int { #1 }
    \str_case_e:nn { \l__affiliations_style_tl }
      {
        {alphabetic}
          {
            \seq_put_right:Nx \l__affiliations_tmpb_seq
                              { \int_to_alph:n{ \int_eval:n {#1 + 1} } }
           }
        {numeric}
          {
            \seq_put_right:Nx \l__affiliations_tmpb_seq
                              { \int_eval:n {\l__affiliations_tmpa_int + 1} }
          }
        {circled}
          {
            \seq_put_right:Nx \l__affiliations_tmpb_seq
                              { \prop_item:Nn \l__affiliations_icons_prop
                                              { \l__affiliations_tmpa_int} }
          }
        {none} { }
      }
  }
```

(*End of definition for* `\__affiliations_return_afil_text:n`.)

__affiliations_output_authors: A helper macro to output the list of authors, with affiliation marks (if any).

```
\cs_new:Nn \__affiliations_output_authors:
  {
    \seq_clear:N \l__affiliations_tmpa_seq
    \prop_map_inline:Nn \l__affiliations_output_prop
      {
        \seq_clear:N \l__affiliations_tmpb_seq
        \clist_map_function:nN {##2} \__affiliations_return_afil_text:n
        \tl_set:Nn \l__affiliations_tmpb_tl
          {
            \seq_use:Nn \l__affiliations_tmpb_seq
                        {\l__affiliations_indices_separator_tl}
          }
        \str_case_e:nn { \l__affiliations_orcid_place_tl }
          {
            {none}
            {
              \seq_put_right:Nx \l__affiliations_tmpa_seq
                {
                  \tl_rescan:nn {} {##1}
                  \exp_not:N
                    \textsuperscript{\tl_use:N \l__affiliations_tmpb_tl}
                }
            }
            {before}
            {
              \seq_put_right:Nx \l__affiliations_tmpa_seq
                {
                  \exp_not:N \__affiliations_recover_orcid:n { ##1 }
                  \tl_rescan:nn {} {##1}
                  \exp_not:N
                    \textsuperscript{\tl_use:N \l__affiliations_tmpb_tl}
                }
            }
            {after}
            {
              \seq_put_right:Nx \l__affiliations_tmpa_seq
                {
                  \tl_rescan:nn {} {##1}
                  \exp_not:N \__affiliations_recover_orcid:n { ##1 }
                  \exp_not:N
                    \textsuperscript{\tl_use:N \l__affiliations_tmpb_tl}
                }
            }
          }
      }
    \seq_use:Nnnn \l__affiliations_tmpa_seq
                  {\l__affiliations_separator_between_two_tl}
                  {\l__affiliations_separator_between_mult_tl}
                  {\l__affiliations_separator_between_last_two_tl}
  }
```

(*End of definition for* `\__affiliations_output_authors:`.)

\affiliations_resolve:n The main macro.

```
\cs_new:Npn \affiliations_resolve:n #1
  {
    \tl_set:Nn \l__affiliations_tmpc_tl { #1 }
    \tl_replace_all:NnV \l__affiliations_tmpc_tl
                        { \and }
                        \l__affiliations_input_names_sep_tl
    \seq_set_split:NVV \l__affiliations_names_seq
                       \l__affiliations_input_names_sep_tl
                       \l__affiliations_tmpc_tl
    \seq_map_inline:Nn \l__affiliations_names_seq
      {
        \tl_clear_new:N \l__affiliations_names_tmp_tl
        \tl_set:Nn \l__affiliations_names_tmp_tl { ##1 }

```

Regex-parsing: We store the {⟨*affiliations*⟩} found in `\affiliation` to a separate sequence, and the {⟨*orcid*⟩} found in `\orcid` to another sequence.

```

        \regex_extract_once:nnN
          {\c{affiliation} \cB. (\c[^BE].*) \cE.}
          { ##1 }
          \l__affiliations_tmp_affil_seq

        \regex_extract_once:nnN
          {\c{orcid} \cB. (\c[^BE].*) \cE.}
          { ##1 }
          \l__affiliations_tmp_orcid_seq

```

Now strip all instances of `\affiliations`{⟨*list*⟩} and `\orcid` {⟨*id*⟩} to receive the name of the author. Also trim all leading and trailing spaces that remain after affiliation and ORCiD replacement.

```

        \regex_replace_all:nnN {\c{orcid} \cB. (\c[^BE].*) \cE.}
                               {}
                               \l__affiliations_names_tmp_tl
        \regex_replace_all:nnN {\c{affiliation} \cB. (\c[^BE].*) \cE.}
                               {}
                               \l__affiliations_names_tmp_tl

        \tl_trim_spaces:N \l__affiliations_names_tmp_tl

```

And store the data in two separate property lists.

```

        \prop_put:Nxx \l__affiliations_tmpa_prop
        { \tl_use:N \l__affiliations_names_tmp_tl }
        { \seq_item:Nn \l__affiliations_tmp_affil_seq {2} }

        \prop_put:Nxx \l__affiliations_orcids_prop
        { \tl_use:N \l__affiliations_names_tmp_tl }
        { \seq_item:Nn \l__affiliations_tmp_orcid_seq {2} }
      }
    \bool_if:NTF \l__affiliations_output_affiliation_bool
```

```
{
  \bool_if:NTF \l__affiliations_output_grouped_bool
    {
      \prop_map_function:NN \l__affiliations_tmpa_prop
                            \__affiliations_resolve_affiliations:
      \group_begin:
        \noindent
        \parbox {\linewidth}
          {
            \__affiliations_output_authors_paragraph_format:
            \__affiliations_output_authors_font:
            \__affiliations_output_authors:
          }
      \group_end:\smallskip\\
      \group_begin:
        \parbox {\linewidth}
          {
            \__affiliations_output_affiliation_paragraph_format:
            \__affiliations_output_affiliation_font:
            \__affiliations_output_affiliations:
          }
      \group_end:
    }
    {
      \seq_clear:N \l__affiliations_tmpa_seq
      \prop_map_inline:Nn \l__affiliations_tmpa_prop
        {
          \str_case_e:nn { \l__affiliations_orcid_place_tl }
            {
              {none}
              {
                \seq_put_right:Nx \l__affiliations_tmpa_seq
                  {
                    \group_begin:
                      \exp_not:N \__affiliations_output_authors_font:
                      \tl_rescan:nn {} {##1}
                    \group_end:\\[0.5ex]
                    \group_begin:
                      \exp_not:N \__affiliations_output_affiliation_font:
                      \tl_rescan:nn {} {##2}
                    \group_end:
                  }
              }
              {before}
              {
                \seq_put_right:Nx \l__affiliations_tmpa_seq
                  {
                    \group_begin:
                      \exp_not:N \__affiliations_output_authors_font:
                      \exp_not:N \__affiliations_recover_orcid:n { ##1 }
                      \tl_rescan:nn {} {##1}\\[0.5ex]
                    \group_end:
                    \group_begin:
                      \exp_not:N \__affiliations_output_affiliation_font:
```

```
                        \tl_rescan:nn {} {##2}
                      \group_end:
                    }
                }
                {after}
                {
                  \seq_put_right:Nx \l__affiliations_tmpa_seq
                    {
                      \group_begin:
                        \exp_not:N \__affiliations_output_authors_font:
                        \tl_rescan:nn {} {##1}
                        \exp_not:N \__affiliations_recover_orcid:n { ##1 }
                        \\[0.5ex]
                      \group_end:
                      \group_begin:
                        \exp_not:N \__affiliations_output_affiliation_font:
                        \tl_rescan:nn {} {##2}
                      \group_end:
                    }
                }
              }
          }
        \noindent
        \seq_use:Nnnn \l__affiliations_tmpa_seq
                      {\l__affiliations_separator_between_two_tl}
                      {\l__affiliations_separator_between_mult_tl}
                      {\l__affiliations_separator_between_last_two_tl}
    }
}
{
  \group_begin:
  \__affiliations_output_authors_font:
  \seq_clear:N \l__affiliations_tmpa_seq
  \prop_map_inline:Nn \l__affiliations_tmpa_prop
    {
      \str_case_e:nn { \l__affiliations_orcid_place_tl }
        {
          {none}
            {
              \seq_put_right:Nx
                \l__affiliations_tmpa_seq
                { \tl_rescan:nn {} {##1} }
            }
          {before}
            {
              \seq_put_right:Nx
                \l__affiliations_tmpa_seq
                  {
                    \exp_not:N \__affiliations_recover_orcid:n { ##1 }
                    \tl_rescan:nn {} {##1}
                  }
            }
          {after}
            {
```

```
                    \seq_put_right:Nx
                      \l__affiliations_tmpa_seq
                        {
                          \tl_rescan:nn {} {##1}
                          \exp_not:N \__affiliations_recover_orcid:n { ##1 }
                        }
                    }
                  }
              }
            \seq_use:Nnnn \l__affiliations_tmpa_seq
                          {\l__affiliations_separator_between_two_tl}
                          {\l__affiliations_separator_between_mult_tl}
                          {\l__affiliations_separator_between_last_two_tl}
            \group_end:
          }
      }
```

(*End of definition for* `\affiliations_resolve:n`.)

`\affiliations_count_authors:n` Count the numbers of authors and saves the result in the global integer variable `\g__affiliations_num_authors_int`.

```
\cs_new:Npn \affiliations_count_authors:n #1
  {
    \tl_set:Nn \l__affiliations_tmpc_tl { #1 }
    \tl_replace_all:NnV \l__affiliations_tmpc_tl
                        { \and }
                        \l__affiliations_input_names_sep_tl
    \seq_set_split:NVV \l__affiliations_names_seq
                       \l__affiliations_input_names_sep_tl
                       \l__affiliations_tmpc_tl
    \int_gset:Nn \g__affiliations_num_authors_int
      {  \seq_count:N \l__affiliations_names_seq  }
  }
```

(*End of definition for* `\affiliations_count_authors:n`.)

`\__affiliations_recover_orcid:n` Return the ORCID associated with an author.

```
\cs_new:Npn \__affiliations_recover_orcid:n #1
  {
    \prop_get:NnNTF \l__affiliations_orcids_prop { #1 }
      \l__affiliations_tmpd_tl
      {
        \tl_if_empty:NTF \l__affiliations_tmpd_tl
          % No ORCID present; no action done.
          { }
          {
            \LinkToORCIDinAffiliations{\tl_use:N \l__affiliations_tmpd_tl}
          }
      }
      % No database entry for author; no action done.
      { }
  }
```

(*End of definition for* `\__affiliations_recover_orcid:n`.)

⟨/package⟩