

OASlcs: Instructions for Authors and the oasics-v2019 Class

Dagstuhl Publishing

Schloss Dagstuhl – Leibniz-Zentrum für Informatik
Oktavie-Allee, 66687 Wadern, Germany
publishing@dagstuhl.de

Abstract

This document provides general instructions for authors on how to prepare articles to be submitted to the *OpenAccess Series in Informatics* series and a description of the typesetting process. All OASlcs articles are written in L^AT_EX using the L^AT_EX style class of the series. Starting in 2019, the newest version of this style **oasics-v2019** is used for publications. This document presents the functionalities of this style class, along with the instructions for the authors.

2012 ACM Subject Classification General and reference → Computing standards, RFCs and guidelines

Keywords and phrases L^AT_EX, type setting, style, changelog, oasics-v2019

Supplementary Material The style file, including a sample document, can be found at
<http://drops.dagstuhl.de/styles/oasics-v2019/oasics-v2019-authors.zip>

1 Typesetting instructions – Summary

OASlcs – OpenAccess Series in Informatics is a series of open access high-quality conference proceedings across all fields in informatics established in cooperation with Schloss Dagstuhl. In order to do justice to the high scientific quality of the conferences that publish their proceedings in the OASlcs series, which is ensured by the thorough review process of the respective events, we believe that OASlcs proceedings must have an attractive and consistent layout matching the standard of the series. Moreover, the quality of the metadata, the typesetting and the layout must also meet the requirements of other external parties such as indexing service, DOI registry, funding agencies, among others. The guidelines contained in this document¹ serve as the baseline for authors, editors, and the publisher to create documents that meet as many different requirements as possible.

Please comply with the following instructions when preparing your article for a OASlcs proceedings volume.

Minimum requirements

- Use pdf_latex and an up-to-date L^AT_EX system.
- Use further L^AT_EX packages and custom made macros carefully and only if required.
- Use the provided sectioning macros: `\section`, `\subsection`, `\subsubsection`, `\paragraph`, `\paragraph*`, and `\subparagraph*` (for more details, see Section 2.4).
- Provide suitable graphics of at least 300dpi (preferably in PDF format).
- Use BIB_TE_X and keep the standard style (`plainurl`) for the bibliography.
- Please try to keep the warnings log as small as possible. Avoid overfull `\hboxes` and any kind of warnings/errors with the referenced BIB_TE_X entries.
- Use a spellchecker to correct typos.

¹ Since the mentioned requirements are constantly changing, these author guidelines but also the OASlcs style are continuously adapted.



Mandatory metadata macros

Please set the values of the metadata macros carefully since the information parsed from these macros will be passed to publication servers, catalogues and search engines. Avoid placing macros inside the metadata macros. For a more detailed explanation and a full list of all metadata macros, see Section 2.3. The following metadata macros/environments are mandatory:

- `\title` and, in case of long titles, `\titlerunning`.
- `\author`, one for each author, even if two or more authors have the same affiliation.
- `\authorrunning` and `\Copyright` (concatenated author names)
The `\author` macros and the `\Copyright` macro should contain full author names (especially with regard to the first name), while `\authorrunning` should contain abbreviated first names.
- `\ccsdesc` (ACM subject classification²).
- `\keywords` (a comma-separated list of keywords).
- `\relatedversion` (if there is a related version like a “full version”, “extended version”, or also a predecessor); please make sure to provide a persistent URL, e.g., at arXiv.
- `\begin{abstract}... \end{abstract}`.

Please do not ...

Generally speaking, please do not override the `oasics-v2019`-style defaults. To be more specific, a short checklist also used by Dagstuhl Publishing during the final typesetting is given below. In case of **non-compliance** with these rules Dagstuhl Publishing will remove the corresponding parts of L^AT_EX code and **replace it with the `oasics-v2019` defaults**. In serious cases, we may reject the L^AT_EX-source and expect the corresponding author to revise the relevant parts.

- Do not use a different main font. (For example, the `times` package is forbidden.)
- Do not alter the spacing of the `oasics-v2019.cls` style file.
- Do not use `enumitem` and `paralist`. (The `enumerate` package is preloaded, so you can use `\begin{enumerate}[(a)]` or the like.)
- Do not use “self-made” sectioning commands (e.g., `\noindent{\bf My Paragraph}`).
- Do not hide large text blocks using comments or `\iffalse ... \fi` constructions.
- Do not use conditional structures to include/exclude content. Instead, please provide only the content that should be published – in one file – and nothing else.
- Do not wrap figures and tables with text. In particular, the package `wrapfig` is not supported.
- Do not change the bibliography style. In particular, do not use author-year citations. (The `natbib` package is not supported.)

This is only a summary containing the most relevant details. Please read the complete document for all details and don’t hesitate to contact Dagstuhl Publishing (`publishing@dagstuhl.de`) in case of questions or comments.

Upholding the above-mentioned minimum formatting requirements will make the job of authors, volume editors, and Dagstuhl Publishing staff easier and will lead to the timely publication of volumes. There is no reason to tweak anything to save space in the published version of the papers.

² <https://submission.dagstuhl.de/services/acm-subject-classification>

Contents

1	Typesetting instructions – Summary	1
2	General Instructions for Authors	4
2.1	Preparation	4
2.2	The <code>oasics-v2019</code> class	4
2.2.1	Paper format	5
2.2.2	Language	5
2.2.3	Input encoding	6
2.2.4	Fonts	6
2.2.5	Colors	6
2.2.6	Spacing and page length	6
2.2.7	Main elements in the document	7
2.3	Top Matter: Authors, affiliations, and other metadata	7
2.4	Structuring the document	9
2.5	Figures and Tables	10
2.5.1	Graphics	12
2.5.2	Tables	12
2.5.3	Rotating floats	12
2.6	Lists, enumerations, and descriptions	12
2.7	Theorem-like environments	12
2.8	Equations / Mathematical formulas	14
2.9	Listings and Algorithms	15
2.10	Bibliography and References	16
2.11	Line numbers	16
2.12	Packaging and Submission	16
3	Publication Workflow and Final Typesetting	17
3.1	Submission Workflow	17
3.2	Final Typesetting	18
A	Packages used in <code>oasics-v2019</code>	20
B	Changes in <code>oasics-v2019</code> compared to <code>oasics-v2018</code>	20
B.1	How to switch from <code>oasics-v2018</code> to <code>oasics-v2019</code>	21
B.2	How to switch from <code>oasics-v2016</code> to <code>oasics-v2019</code>	22
C	<code>oasics-v2019</code> styles of lists, enumerations, and descriptions	23
D	<code>oasics-v2019</code> theorem-like environments	23

2 General Instructions for Authors

On the following pages, the most important instructions for authors are provided to prepare a document to be published in a *OpenAccess Series in Informatics* series volume.

2.1 Preparation

To prepare an article for the OASlcs series, the authors have to provide a \LaTeX ³ document using the `oasics-v2019` style. An alternative workflow using another format like Microsoft Word is not supported. Along with \LaTeX , \BibTeX has to be used to prepare the bibliography (see Section 2.10).

We suggest to employ a recent \TeX installation: the most important distributions, \TeX Live, \MiKTeX / \proTeXt and \MacTeX , all provide at least 2017 versions. Older versions should (in principle) work as well. Besides \LaTeX and \BibTeX , no other processing tools and packages with external dependencies like `biber`, `minted`, `pygmentize`, etc. are supported.

2.2 The `oasics-v2019` class

The `oasics-v2019` class assists authors in preparing articles for *OpenAccess Series in Informatics* with \LaTeX . It adapts \LaTeX 's standard `article` class to meet some requirements for OASlcs and provides a specific layout.

The package⁴ consists of the following files:

- `oasics-v2019-authors-guidelines.pdf` (this documentation),
- `oasics-v2019-sample-article.tex` (the \LaTeX master file, to be used as a template),
- `cc-by.pdf` (logo for Creative Commons CC BY),
- `oasics-logo-bw.pdf` (logo for OASlcs),
- `orcid.pdf` (logo for ORCID), and
- `oasics-v2019.cls` (the \LaTeX class file, providing adaptations of `articles.cls` for OASlcs and producing the layout).

To use `oasics-v2019`, put `oasics-v2019-sample-article.tex`, `oasics-v2019.cls`, `cc-by.pdf`, `oasics-logo-bw.pdf`, and `orcid.pdf` in your working directory, edit the file `oasics-v2019-sample-article.tex` in your preferred text editor and run \LaTeX as usual. (See the following section for more detailed advise.)

`oasics-v2019` tries to benefit as far as possible from standard \LaTeX packages⁵. Therefore, it should also be easy to compile an already written manuscript with the `oasics-v2019` layout. To learn more about the underlying packages, we refer to their documentations (try e.g., `texdoc [package name]` at your shell prompt or visit tug.ctan.org).

³ This documentation is not intended to give an introduction to \LaTeX . For questions concerning \TeX systems/installations or the \LaTeX mark-up language in general please visit www.tug.org, www.dante.de, uk.tug.org or any other \TeX user group worldwide. The essential reference for \LaTeX is *Mittelbach F., Goossens M. (2004) The \LaTeX Companion. 2nd edn.* [1], but there are many other good books delivering insight into \LaTeX .

⁴ Available on <http://drops.dagstuhl.de/styles/oasics-v2019/oasics-v2019-authors.zip>.

⁵ A list of the used packages is provided in section A in the Appendix.

2.2.1 Paper format

For OASiCs, the A4 format is the mandatory paper format and will be used to prepare the final version. The option `a4paper` must be inserted in the optional argument of `\documentclass`. For intermediate versions, the US-letter format `letterpaper` can be used alternatively.

Tip: You can add the option `draft` as argument of `\documentclass` to highlight overfull hboxes. This will also temporarily replace figures with placeholders, which might speed up the compilation process.

Options: Table 1 shows the most common options for arguments of `\documentclass`. Some of the options are explained in later sections: `USenglish` (Section 2.2.2), `UKenglish` (Section 2.2.2), `numberwithinsect` (Section 2.7), `cleveref` (Section 2.10), `autoref` (Section 2.10), `anonymous` (Section 2.3), and `authorcolumns` (Section 2.3).

■ **Table 1** Common Options for `\documentclass`.

Option	Description
<code>a4paper</code>	A4 paper format
<code>letterpaper</code>	US-letter format
<code>draft</code>	Highlight overfull hboxes
<code>USenglish</code>	Enables US-English specific typographical rules
<code>UKenglish</code>	Enables UK-English specific typographical rules
<code>numberwithinsect</code>	Enables section-numbered lemma etc.
<code>cleveref</code>	Enables support for the <code>cleveref</code> package
<code>autoref</code>	Enables support using <code>autoref</code> provided by the <code>hyperref</code> package
<code>anonymous</code>	Enables support for double-blind peer-review
<code>thm-restate</code>	Enables support for the <code>thm-restate</code> package for restating theorems
<code>authorcolumns</code>	Enables two-column displaying of author details (only applicable for >6 authors)

2.2.2 Language

The document language is chosen in the optional argument of the `\documentclass` command in the L^AT_EX master file. This is necessary to influence the “culturally-determined typographical (and other) rules” (like hyphenation) within the document. Possible values are `USenglish`, `UKenglish` and others⁶.

Tip: Older version of the used L^AT_EX package `babel` might not know the default language (`UKenglish`). For many installations, setting `english` as the document language solves the problem: `\documentclass[english]{oasics-v2019}`. Please note that out-dated language options such as `english` will be replaced by the OASiCs default `UKenglish` during the finalization process.

⁶ This depends on the L^AT_EX package `babel`. For further details, please visit the documentation of the package at <https://www.ctan.org/pkg/babel>.



2.2.3 Input encoding

oasics-v2019 preselects UTF-8 as input encoding. Please do not change the input encoding because otherwise the volume compilation might become difficult.

2.2.4 Fonts

oasics-v2019 uses the Latin Modern font family. This is a recent redesign of the good old Computer Modern fonts. Latin Modern provides many characters and all necessary mathematical fonts. If your T_EX installation does not provide the Latin Modern family, Computer Modern is used as a fallback. Do not use a different main font. For example, the usage of the times package is forbidden.






oasics-v2019 preloads the package amssymb to make additional mathematical symbols available. Other symbol packages, e.g., stmaryrd, may be added. Moreover, the script mathematical alphabet is provided by loading the eucal package. Please avoid loading the MnSymbol package. (Note that oasics-v2019 removes all symbol re-definitions done by MnSymbol anyhow, but would retain its newly introduced symbol definitions.)

2.2.5 Colors

In the OASlcs style, very few colours are used for purely aesthetic purposes. To increase readability especially for color-blind persons, we recommend not to use colors for emphasizing text. Instead of this other mechanisms like \textbf, \emph or dashed boxes (e.g. provided by dashbox⁷ package) can be used.

Table 2 lists all colors that are pre-defined and used in oasics-v2019. To reuse one of the pre-fined colors, just use the name given in the first column of the table (e.g. \textcolor{oasicsGray}).

■ Table 2 Pre-defined colors in oasics-v2019.

Name	Color	rgb Code	Usage
oasicsGray		0.31, 0.31, 0.33	heading in top matter, enumerations
oasicsBulletGray		0.60, 0.60, 0.61	bullet points, footnoterule
oasicsLineGray		0.51, 0.50, 0.52	line in heading of abstract and bibliography
oasicsLightGray		0.85, 0.85, 0.86	background of listings
oasicsYellow		0.99, 0.78, 0.07	section numbers, figure and table captions

2.2.6 Spacing and page length

oasics-v2019 provides a liberal spacing to improve the readability of the articles. Please do not modify the default spacing and also do not use any packages (e.g. savetree) to manipulate the OASlcs default spacing. It will be reset to default during the final typesetting process by the OASlcs office. To avoid any problems due to the OASlcs liberal spacing, it is

⁷ <https://ctan.org/pkg/dashbox>

\dashbox

~~savetree~~

highly recommended for the editors of OASICS volumes to set a flexible page limit⁸ (e. g., of 12–15 pages) and to explicitly exclude the bibliography, the first page(s) containing the top matter (authors, affiliation, keywords, abstract, ...) and a brief appendix (of up to 5 pages) from this page limit.

During the preparation, the authors should take care of suitable linebreaks and pagebreaks. For instance, no overfull `\hboxes` should occur in the warnings log.

`\hbox`

2.2.7 Main elements in the document

The body of a OASICS document consists in general of the following elements:

- top matter containing the metadata describing the document (see Section 2.3);
- section headings (see Section 2.4);
- figures and tables (see Section 2.5);
- lists (see Section 2.6);
- theorem-like environments like lemma, definitions, remarks, and proofs (see Section 2.7);
- mathematics, both in text and in display (see Section 2.8);
- listings and algorithms (see Section 2.9);
- references (see Section 2.10);
- and text.

2.3 Top Matter: Authors, affiliations, and other metadata

The term “top matter” is used in this document for the metadata information found at the beginning of an article. It includes information that would appear in a bibliographic reference to the paper, plus information to clearly identify the authors, funding information, acknowledgements, and the abstract. The top matter also includes information such as subject classification and keywords that simplifies the indexation of the articles. Some elements are required, while others are optional (each element is described in detail below).

The header of a OASICS article is the only part where some specific commands/environments are required:

- The title is tagged as usual with the `\title{...}` command. If you need a short form for the running head, use the optional `\titlerunning{...}`. The title (as well as headings in the document) should be capitalized (see Section 2.4 for further details). `\title`
`\titlerunning`
- Authors and their affiliations are rendered blockwise for OASICS⁹. The information for each author has to be tagged with a redefined `\author` macro, which is structured as follows: `\author{name}{affil}{email}{orcid}{funding}`. While the parameters `name` and `affil` are mandatory, the parameters `email`, `orcid`, and `funding` are optional and can be empty. `\author`
- For each author at least one affiliation has to be provided. It is up to the author to decide about the level of detail of the affiliation information, but at least the main affiliation name, the city¹⁰, and country have to be provided. Further details like department or postal address are optional. If several affiliations should be provided for one author, please use the `\and` macro to separate the different affiliations. Authors without an `\and`

⁸ Please note that we expect a certain flexibility at any page limit (+/- 1 page) to avoid too much extra effort to harmonize OASICS-compliant formatting and page limit for both authors and final typesetting by the OASICS Office.

⁹ This has changed in 2017; the formerly used `authblk` mechanism to output authors and affiliations in footnote style is deprecated; the `\affil` command is obsolete for the `oasics-v2019` class.

¹⁰ The city is only mandatory if it is not part of the affiliation name.

affiliation should at least indicate their place of residence and their country of residence. An (optional) URL of an author's personal webpage should simply be added to the affiliation with `\and`, for instance

```
\author{John Q. Public}
{Dummy University Computing Laboratory, [optional: Address], Country \and
My second affiliation, Country \and \url{http://www.myhomepage.edu}}
{johngpublic@dummysuni.org}
{https://orcid.org/0000-0002-1825-0097}
{[optional] author-specific funding acknowledgements}
```

- The usage of an ORCID is currently optional but highly recommended. For more details regarding ORCID, please visit <https://orcid.org/>. If you don't have an ORCID yet, the registration is very quick and simple (see <https://orcid.org/register>). Please enter the full https URL including hyphens between every fourth digit as recommended by ORCID¹¹. In the PDF, the ORCID symbol that links to the ORCID profile is shown behind the respective author. With the next upgrade of *Dagstuhl Research Online Publication Server (DROPS)*, ORCIDs will also become part of the machine readable metadata set.
- Per default, author details (name, affiliation, ...) are displayed in one column under each other. In case of many authors, this might result in a lot of whitespace (especially if the authors have only short affiliations). To avoid this, it is possible to activate a two column style for the authors by adding `authorcolumns` as argument of the `\documentclass`:
`\documentclass[authorcolumns]{oasics-v2019}`. Please note that this option is only available in case of more than 6 authors.
- There is a mandatory `\authorrunning{...}` macro for the running header concatenating all author names (first names have to be abbreviated).
- `\Copyright{...}` has just one argument specifying the copyright holder that is usually the team of authors. Please concatenate the full names of all authors¹².
- To hide all author related information e.g. for double-blind peer review, please add `anonymous` as argument of the `\documentclass`:
`\documentclass[anonymous]{oasics-v2019}`.
- `\ccsdesc{...}` is for classification information following the ACM 2012 Computing Classification System; it is required, too. Please use one `\ccsdesc` command per classification. If the command is used several times, all specified classifications are concatenated in the resulting PDF (using semicolons as auto-generated delimiters). The `\subjclass` command used in previous versions OASlcs L^AT_EX style is obsolete. Please choose 1–3 ACM 2012 classifications from <https://submission.dagstuhl.de/services/acm-subject-classification>.
- `\keywords{...}` must be used to capture keywords. Please use commas as delimiters. The first word of the list and every proper noun should be capitalized. All other items should be lowercase.
- `\category{...}` may be used to provide category information. This macro should only be used for “special” papers (like “invited papers”) and not for regular articles. Please consult the responsible editor of the volume.

¹¹ <https://support.orcid.org/knowledgebase/articles/116780>

¹² Please add “serial comma” also known as “Oxford comma” before “and”.

- `\relatedversion{...}` may be used to denote a related version like a “full version”, “extended version”, or also a predecessor usually published in a repository like arXiv or HAL. As all metadata should be self-contained¹³, please add a persistent URL¹⁴ to this version in this macro, e.g. `\relatedversion{A full version of the paper is available at \url{...}}.` This also simplifies the access for all readers. Additional to the URL, you might add a reference (`\cite{...}`). `\relatedversion`
- `\supplement{...}` may be used to denote supplementary material like related research data, source code, ... slides hosted on a repository¹⁴ like zenodo¹⁵, figshare¹⁶, ..., Software Heritage¹⁷. `\supplement`
- `\funding{...}` may be used to capture a general funding statement. Note that we recommend to provide those kind of funding acknowledgements that can be clearly assigned to a specific author in the corresponding `\author` macro. In the PDF document, all the funding information is displayed in a coherent block, where the author-specific details will automatically be preceded by the name of the author. `\funding`
- `\acknowledgements{...}` may be used to capture an acknowledgement. `\acknowledgements`

The commands mentioned so far should be used in the document preamble of the \LaTeX file. Providing a title, at least one author (incl. `\authorrunning`), copyright information, subject classification, and keywords is required.

To typeset an abstract use `\begin{abstract}...\end{abstract}`. The environment `abstract` must be placed after `\begin{document}` and `\maketitle`!

Note that subject classifications, keywords, ..., and acknowledgements will be rendered together with the abstract. So it is necessary to use the `abstract` environment in order to get the output for `keywords`, `ccsdesc` etc.

All the copyright information (and a possible DOI specification) on the first page can be hidden using the `\hideOASICS` command. This might be useful to prepare a document using the OASICS style, but which is not intended to be published within the series (e.g. for an extended version to be published on arXiv). `\hideOASICS`

2.4 Structuring the document

To organize the content and to structure the document, the default \LaTeX commands for headings have to be used. At least the first three levels of section headings have to be numbered to support the internal referencing of the respective sections and also to simplify referencing in other documents. Table 3 provides an overview of the provided heading commands. All headings, including the title, have to be left aligned and should be capitalized.

Please do not revise the provided commands and e.g., modify text size or spacings. Moreover, please do not add self-defined commands to structure your document. The OASICS office will reset self-defined heading styles to the default, for instance `\medskip\noindent\textbf{Some subparagraph.}` will be replaced by `\subparagraph*{Some subparagraph.}` `\section`
`\subsection`
`\subsubsection`
`\paragraph`
`\paragraph*`
`\subparagraph`

¹³ Metadata should be self-contained as they are not only part of the document / PDF but also extracted and stored in a machine-readable format along with the actual document.

¹⁴ Please note: As hosting on a (personal or university) webpage or in cloud storage is not really sufficient for durable / persistent file storage, we highly recommend you to publish your document in a reliable repository like arXiv or HAL. This recommendation also applies to supplementary materials such as data, software, posters, ..., slides. Please store them in repositories such as zenodo or Software Heritage.

¹⁵ <https://zenodo.org/>

¹⁶ <https://figshare.com/>

¹⁷ <https://www.softwareheritage.org/>

Capitalization. Headings and the title have to be capitalized: The first word (and usually the last word) is capitalized, as well as all major words. Minor words are in lower case. For a detailed description, please use the Chicago Style Guideline or use a web service, such as <http://individed.com/code/to-title-case/>.

■ **Table 3** Headings

Command	Level	Example	Font size and style
<code>\title</code>	0	This is the Title	17.28pt, bold
<code>\section{...}</code>	1	1 Section	12pt, bold
<code>\subsection{...}</code>	2	1.1 Subsection	12pt, bold
<code>\subsubsection{...}</code>	3	1.1.1 Subsubsection	12pt, bold
<code>\paragraph{...}</code>	4	1.1.1.1 Paragraph	10.5pt, bold
<code>\paragraph*{...}</code>	4	Paragraph	10.5pt, bold
<code>\subparagraph*{...}</code>	5	Subparagraph. Some text after	10pt, bold

Referencing. L^AT_EX provides an easy mechanism to reference sections and also almost everything else that is numbered (e.g., figures, tables, algorithms, ...). To use this mechanism, all referenced items should have a marker (a.k.a. `\label{key}`) and instead of direct textual references, the command `\ref{key}` should be used within the L^AT_EX code.

It is up to the authors to use the extended functionalities provided by either the `cleveref`¹⁸ package or the `\autoref{}` command, which is part of the `hyperref`¹⁹ package. To use these functionalities, please add `cleveref` and/or `autoref` as argument of the `\documentclass`: `\documentclass[cleveref, autoref]{oasics-v2019}`. This is explicitly necessary for using these mechanisms with theorem-like environments. Otherwise all theorems, lemmas, corollaries, ... will be referenced as theorems (see also Section 2.7).

2.5 Figures and Tables

Figures, tables, and all other floating objects (like algorithms) can be placed at the top or bottom of a document page (using the placement option `t` or `b`, e.g., `\begin{figure}[t]`). Alternatively they can be placed on a separate page using the placement option `p`. The top position is the preferred position. The fixed position in the document (placement option `h`) should only be used rarely as small changes in the document might result in a completely different layout.

Wrapping of text around figures and tables is not supported. So don't use packages like `wrapfig`.

Use captions, labels, and references for all floating objects explicitly (e.g., “as depicted in Figure `\ref{fig:example}`”) and not implicitly (e.g., “as depicted in the figure below”). Figure captions have to be placed below the figures. Table captions (and also captions of other text-like floating environments like listings and algorithms) have to be placed above the table. Captions are always left aligned.

¹⁸<https://ctan.org/pkg/cleveref>

¹⁹<https://ctan.org/pkg/hyperref>

For small/related figures place figures as sub-figures within one figure. At least the main figure should have a textual caption. The preloaded `subcaption`²⁰ package provides the `\subcaption` command to add sub-captions. Alternatively it provides `subfigure` and `subtable` environments. Please do not load the deprecated `subfig` package. The following fragment shows the code resulting in Figure 1:

`\subcaption`
`subfigure`
`subtable`

`oasics-logo-bw.pdf`

(a) Subfig 1.

`oasics-logo-bw.pdf`

(b) Subfig 2.

■ **Figure 1** Just an Example.

■ **Listing 1** L^AT_EX code of Figure 1.

```
\begin{figure}
  \begin{subfigure}[t]{0.5\textwidth}
    \centering
    \includegraphics[draft, width=0.3\textwidth]{oasics-logo-bw}
    \caption{Subfig 1.}
  \end{subfigure}\hfill
  \begin{subfigure}[t]{0.5\textwidth}
    \centering
    \includegraphics[draft, width=0.3\textwidth]{oasics-logo-bw}
    \caption{Subfig 2.}
  \end{subfigure}
  \caption{Just an Example.}\label{fig:example}
\end{figure}
```

Figures and tables have to be centered. For figures and tables placed within a floating environment, use `\centering`. For figures and tables placed directly within the text, use the `center` environment.

`\centering`
`center`

It is highly recommended to use vector graphics whenever possible. Alternatively authors should provide figures with a high resolution (300 dpi or more).

Hint. If your figure consists mainly of text, we highly recommend to frame the figure (e.g. using the command `\fbox`). This significantly helps to separate the content of the figure from the adjacent main text. Figure 2 shows an example for such a framed figure. The according L^AT_EX code is depicted in Listing 2.

`\fbox`

Lorem ipsum dolor sit amet, consetetur
sadipscing elitr, sed diam nonumy eir-
mod tempor invidunt ut labore et dolore
magna aliquyam erat, sed diam volup-
tua. At vero eos et accusam et justo
duo dolores et ea rebum. Stet clita kasd
gubergren, no sea takimata sanctus est
Lorem ipsum dolor sit amet.

■ **Figure 2** Example for framed figure with mainly textual content.

■ **Listing 2** L^AT_EX code of Figure 2.

```
\begin{figure}
  \centering
  \fbox{
    \begin{minipage}{0.9\textwidth}
      Lorem ipsum dolor sit amet ...
    \end{minipage}
  }
  \caption{Example for framed ...}
  \label{fig:framedExample}
\end{figure}
```

²⁰<https://ctan.org/pkg/subcaption>

2.5.1 Graphics

`\include-
graphics
\graphicspath`

The standard interface for graphic inclusion is the `\includegraphics` command provided by the `graphicx` package. Note that the `\graphicspath` command allows authors to declare one or more folders in which the `graphicx` package looks for the image files; so providing the path with each `\includegraphics` command is not necessary.

2.5.2 Tables

Preloaded packages are: the `array`²¹ package (for introducing new column types), the `multirow`²² package (row spanning cells) and the `tabularx`²³ package (automatic column width calculation).

The `threeparttable` package is preloaded to ease the use of table footnotes. Please read the short documentation in `threeparttables.sty`²⁴ to see how the related commands are applied.

2.5.3 Rotating floats

`sidewaysfigure
sidewaystable`

The preloaded `rotating` package provides the two environments `sidewaysfigure` and `sidewaystable`. They allow for the rotation of floating objects.

2.6 Lists, enumerations, and descriptions

`itemize
enumerate
description`

Other elements that provide structured information within L^AT_EX documents are bulleted lists (also known as `itemize` environments), enumerations (a.k.a. `enumerate` environments), and descriptions. List labels are set flush left.

The `enumerate`²⁵ package is preloaded, so you can use `\begin{enumerate}[(a)]` or the like. `oasics-v2019` provides several predefined environments, namely `itemize`, `enumerate`, `alphaenumerate`, `romanenumerate`, `bracketenumerate`, and `description`. Examples of the different styles are depicted in Section C in the appendix.

~~`enumitem
paralist`~~

Please do not use other packages like `enumitem` or `paralist` to revise `itemize` or `enumerate` environments, as they change the predefined layout of all related environments and therefore will be removed by Dagstuhl Publishing in the final typesetting process.

2.7 Theorem-like environments

`theorem
lemma
corollary
definition
proposition
exercise
example
note
note*
remark*
remark
claim
claim*`

To display theorem-like fragments appropriately, the `amsthm`²⁶ package is preloaded, and the following environments are already introduced: `theorem`, `lemma`, `corollary`, `definition`, `proposition`, `exercise`, `example`, `note`, `note*`, `remark*`, `remark`, `claim`, and `claim*`. For the predefined environments, a consecutive numbering is applied, e.g., all predefined theorem-like environments use the same counter. Examples of the different styles are depicted in Section D in the appendix.

Setting up additional environments works with the `\newtheorem` mechanism from the `amsthm` package. For example, add to your document preamble

```
\newtheorem{conjecture}[theorem]{Conjecture}
```

²¹<https://www.ctan.org/pkg/array>

²²<https://www.ctan.org/pkg/multirow>

²³<https://www.ctan.org/pkg/tabularx>

²⁴<https://www.ctan.org/pkg/threeparttable>

²⁵<https://www.ctan.org/pkg/enumerate>

²⁶<https://www.ctan.org/pkg/amsthm>

See also the `amsthm` package documentation.

Available `\theoremstyle`s are: `plain`, `definition`, `remark`, and `claimstyle` (the first three are from the `amsthm` package, but slightly modified for OASICs).

Note that for OASICs it is recommended that all numbered theorem-like environments use one and the same counter, i. e., the counter of the default environment `theorem`.

By default, theorem-like environments are numbered consecutively throughout the document. To number the environments subordinately within sections use the class option `numberwithinsect`: `\documentclass[numberwithinsect]{oasics-v2019}`.

number-
withinsect

Additional to the above mentioned numbered environments, `oasics-v2019` also provides two unnumbered proof environments: `proof` and `claimproof`. Both differ only in the delimiting QED symbol; \blacktriangleleft is automatically appended at the end of `proof` environments; \triangleleft is appended at the end of `claimproof` environments.

To repeat a theorem-like environment, the style supports the package `thm-restate`, which is provided as part of the `thmtools`²⁷ bundle. To activate this support, please add `thm-restate` as argument of the `\documentclass`:

thm-restate

`\documentclass[thm-restate]{oasics-v2019}`.

For further information, please see the `thmtools` documentation.

Hint 1. In difference to the style provided by `amsthm` package, the body of the `definition` environment is formatted as in `theorem` environments, e. g. italic text and not upright text as in the original environment. This should help to separate text of the definition from surrounding regular text. If an upright text as in the original environment is preferred, please add an alternative `definition` environment to the document preamble, e. g. as following:

```
\theoremstyle{definition}
\newtheorem{definition2}[theorem]{Definition}
```

Hint 2. Depending on the style of the environment, it might be difficult to clearly identify the end of the respective environment, see following example:

This is just some regular text right before an example environment.

► **Example 1.** This is the first paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

This is another paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

This is regular text after the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

²⁷ <https://www.ctan.org/pkg/thmtools>

`\oasicsEnd`

In this case, some authors prefer to clearly mark the end of the environment. This can be done using the command `\oasicsEnd`. This will add a symbol (`\lrcorner`) add the end of the line (similar a qed symbol), see Example 2 below:

► **Example 2.** This is the first paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

This is another paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. ┘

Hint 3. As mentioned previously, to use `cleveref` and/or `autoref` also with theorem-like environments, please add `cleveref` and/or `autoref` as argument of the `\documentclass`: `\documentclass[cleveref, autoref]{oasics-v2019}`.

2.8 Equations / Mathematical formulas

`$. . . $`
`equation`
`equation*`
`\[. . . \]`

Equations or mathematical formulas can either be in line with the text (using `$. . . $`) or in an equation-type environment (e. g., `\[. . . \]`). The latter use is also known as display math mode. In this display math mode, the separately displayed formulas should be numbered (consecutive numbering is recommended). Important equations (and all large equations) should be placed in a separate block (i.e. in display math mode) and not inline. Furthermore, important equations should be labeled to reference them.

The `amsmath` package is preloaded, and you are encouraged to use the mark-up it provides instead of old-style standards like the `eqnarray` environment or the `\over` command.

~~`$$. . . $$`~~
`\[. . . \]`

Display math formulas. Do not use the T_EX primitive `$$. . . $$` to typeset display math formulas as this might cause problems²⁸, for instance the alignment option is ignored. Instead use the L^AT_EX variant of this environment, namely `\[. . . \]`. During the final typesetting process, the OASlcs office will replace `$$. . . $$` with `\[. . . \]`.

`align, align*`

To avoid inconsistent spacing, do not use the `displaymath`, `eqnarray`, or `eqnarray*` environments because those are not supported by the preloaded `amsmath` package. `\[. . . \]` may be used instead of `displaymath`, while `eqnarray`, and `eqnarray*` may be replaced with `align`, or `align*`.

Display math formulas are aligned left. So do not use environments that ignore/change this alignment of these formulas. Deviating from this rule, the centered setting of inference rules (e.g. using the `mathpartir`²⁹ package) is permitted, however, for reasons of consistency, a left-justified orientation is also recommended here.

²⁸ For more details, visit <https://ctan.org/pkg/l2tabu-english>, [2].

²⁹ <https://www.ctan.org/pkg/mathpartir>

2.9 Listings and Algorithms

An important part in many computer science publications is “pseudo code” or code fragments. In the L^AT_EX ecosystem, several packages exist to format code/algorithms. It is highly recommended to use one of the following packages and not to use self-defined environments: `listings`³⁰, `algorithm`³¹, or `algorithm2e`³².

The `listings` package is preloaded. It provides the `lstlisting` environment to typeset displayed code. Here, the package is configured to produce a grey background for listings. `lstlisting`

The following example shows how to use captions and labels with the `lstlisting` environment:

```
\begin{lstlisting}[caption={Useless code.},label=list:8-6,float=h,
                        abovecaptionskip=-\medskipamount]
for i:=maxint to 0 do
begin
  j:=square(root(i));
end;
\end{lstlisting}
```

The resulting code is depicted in Listing 3.

■ Listing 3 Useless code.

```
for i:=maxint to 0 do
begin
  j:=square(root(i));
end;
```

Also note the `float` option that makes the listing float. Instead of the `caption` option, one might prefer the `title` option, which outputs the argument without the “Listing” label. To globally change the label name from “Listing”, add to your document preamble e.g.,

```
\renewcommand{\lstlistingname}{Algorithm}
```

Please read the package documentation for more information on the `lstlisting` environment and how to adapt it locally.

The other above-mentioned recommended packages to include code or algorithms, `algorithm` and `algorithm2e` are not preloaded. It is up to the authors to load one of these packages. If so, please use the option `ruled` to ensure the correct placement of the caption (as mentioned before, captions of text-like content are placed above the content and left aligned), for instance `\usepackage[ruled]{algorithm}` or `\usepackage[ruled]{algorithm2e}`. For further details, please read the documentation of these packages.

`algorithm`
`algorithm2e`

³⁰<https://www.ctan.org/pkg/listings>

³¹<https://www.ctan.org/pkg/algorithms>

³²<https://www.ctan.org/pkg/algorithm2e>

`\bibliography`
`\biblio-`
`graphystyle`
`{plainurl}`
`\cite`
~~`\nocite{*}`~~

2.10 Bibliography and References

OASlcs uses BibT_EX to format references. Thereby the BibT_EX style `plainurl` is used for BibT_EX processing. Please do not change the bibliographic style. OASlcs only allows for numerical citation and forbids author-year citations. (So the `natbib` package is not used by `oasics-v2019`.)

All entries of the bibliography has to be referenced in the text by using the `cite` macro. So do not use the command `\nocite{*}` to add unreferenced bibliographic entries to the bibliography.

In general, references should be as complete as possible. So the authors should not abbreviate author names or journal names in references and should ensure that all mandatory fields according to the BibT_EX standard are available, e. g., no errors and warnings about the referenced BibT_EX entries should occur. If possible references should contain an external link (e. g., DOI or URL). DOI is preferred.

Hint. The *dblp computer science bibliography*³³ indexes more than 4 million publications from major computer science outlets. A large amount of the indexed publications also come with a DOI. Furthermore, *dblp* provides a BibT_EX export service, so that exported entries can be easily reused in the paper's bibliography. It is highly recommended to use *dblp* to enrich the references and e. g., add missing DOIs.

2.11 Line numbers

The `lineno` package is loaded, and line numbering is switched on by default. To (globally) switch line numbering off, use `\nolinenumbers` (in the document preamble).

2.12 Packaging and Submission

Please provide a self-contained archive file containing all files necessary to compile/process the document. It is mandatory that there is exactly one L^AT_EX file per article (and no include-files for sections/subsection) and one BibT_EX file. All other files like graphics should be part of the archive.

³³<https://dblp.org>; Search: <https://dblp.org/search/>

3 Publication Workflow and Final Typesetting

After the submission of the camera-ready version of the document to the Dagstuhl Publishing submission server, the document will be checked/revised by the editors and finally by Dagstuhl Publishing.

3.1 Submission Workflow

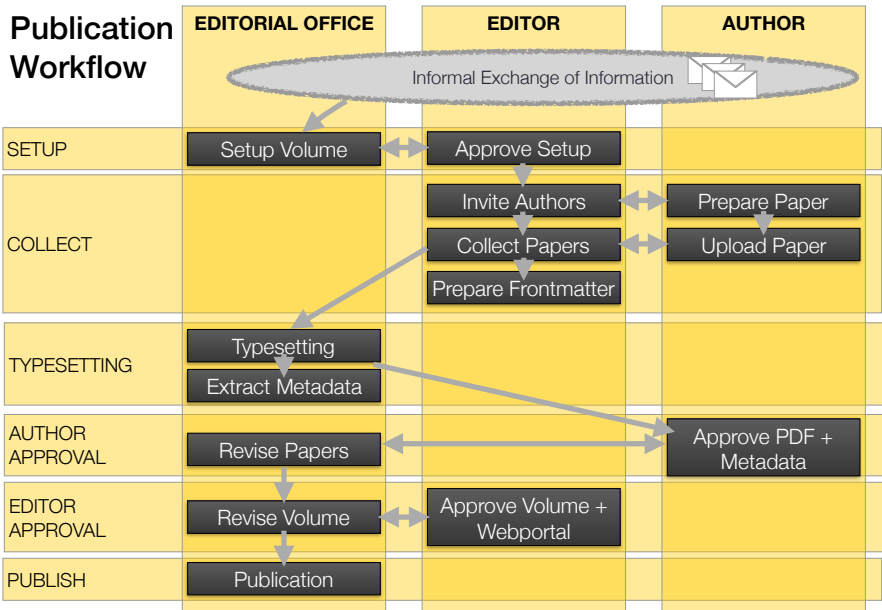


Figure 3 Publication Workflow.

Figure 3 gives an overview of the various steps on the way to publishing a volume/article that take place after the review process. Usually 6–12 months before the planned publication of the volume the OASICs office clarifies some necessary details (e.g. the aimed date of publication and a raw and preliminary schedule). This **informal exchange of information** usually happens by E-mail. The actual **publication workflow** of the volume starts approx. 12–16 weeks before the planned publication (e.g. the conference) and contains the following sub-steps, which all take place in the Dagstuhl Submission System <https://submission.dagstuhl.de>.

1. During the **setup** phase, the OASICs office and the editors determine the key data of a new volume: schedule, editor details, paper categories (such as “Invited Talk”, “Short Paper”, “Regular Paper”) and the author agreement. This step ends up with a web-portal for the volume on the Dagstuhl submission server (being set-up by Dagstuhl Publishing) which guides editors and authors through all further steps.
2. Using this web-portal the editors then **collect** the camera-ready versions of the articles to be published. They inform the authors about the acceptance of their article and invite them to submit their camera-ready version directly to the submission system.
 - a. The authors **prepare** their article according to the instructions provided in this document and upload a zip-archive containing all source files as described in Section 2.12 along with a signed author agreement.

- During the submission, the LaTeX document is compiled on the server.
 - Only correct and complete documents are accepted.
 - After the submission of the LaTeX document, the author is asked for missing (mandatory) metadata.
- b. The editors monitor the submission process and take care that all authors submit complete source files in due time. It is not expected that the editors compile/revise/typeset the provided sources.
- c. After the submission, the editors check the final order of the submitted articles within the submission tool and prepare a draft of the **frontmatter**³⁴.
- 3. After the volume has been submitted to Dagstuhl Publishing by the editors, the OASICs office carefully checks all papers and revises them to achieve compliance with the instructions provided in this document. These checks include the actual **typesetting** of the documents as well as the respective **metadata**. All changes (and open issues, if any) are documented in a changelog provided to the authors in the next step.
- 4. Usually two to three weeks before publication of the volume, the OASICs office asks the **authors for their approval** of the revised documents and the metadata. To avoid any delay of the publication process, this author approval phase is limited to 3–5 days. In order to enable all authors to participate, the exact period will be announced to the authors at least one week in advance. Authors may then also authorize other users to perform the approval for them, if necessary. If the authors (or authorized users) nevertheless allow the approval period to expire, the corresponding article is considered passively approved.
- 5. After a possible revision of the documents and/or their metadata incorporating the feedback of the authors, the OASICs office asks the **editors for approval** of all documents and the complete volume.
- 6. After the editor approval, the OASICs office officially **publishes** the proceedings volume (including DOI registration, registration for long-term archiving, submission to indexing services like dblp or Google Scholar).

All above mentioned sub-steps of the submission process are actively supported by the Dagstuhl Submission System introduced in Spring 2019.

3.2 Final Typesetting

The final typesetting of the document is done by the OASICs office before asking for the author's approval for publication. In this phase, the OASICs office tries to adapt/revise the documents as little as possible, while still achieving a consistent appearance in line with the corporate identity of the OASICs series. Basically, the articles are revised to ensure compliance with the standards described in this document. Nevertheless, the publisher tries to maintain the personal touch of the authors. If the instructions given in this document and the template file (`oasics-v2019-sample-article.tex` provided with the style archive) have been followed closely, only minimal changes will be made to the articles if any.

The internal workflow during the typesetting process is essentially based on the list of requirements given in Section 1. In particular,

- the `\documentclass` and paper-format of the article will be checked;
- individual (non-style-conforming) spacing and font modifications will be removed;

³⁴ The editors only have to prepare the editor details, the preface, and details regarding the conference organisation. Volume-specific information and the final table of contents will be added by the OASICs office during the finalisation of the volume.

- the top matter and mandatory metadata fields will be checked/corrected;
- default sectioning commands will be applied wherever possible and `\texorpdfstring{...}{...}` macros will be inserted for non-ascii headlines;
- default enumeration/itemization/description styles will be applied wherever possible (see Section 2.6 and Appendix C);
- default theorem-like environments and the corresponding proof environments will be used wherever possible (see Sections 2.7 and Appendix D);
- default math environments will be applied wherever possible (see Section 2.8);
- figures, tables and listings/algorithms will be revised to fit the standards described in Section 2;
- in particular, wrapped figures will be converted into ordinary figure-environments;
- all overfull `\hboxes` will be removed;
- quotation marks will be standardized to “...”, dashes will be standardized to “–”;
- deprecated commands will be replaced with current analogues (e.g., `{\em ...}` → `\emph{...}`); see also [2];
- in exceptional cases font-sizes and scaling are revised to improve the readability;
- the `BIBTEX` file is revised by adding missing mandatory fields (according to the `BIBTEX` standard) and enabling or adding URL/DOIs where applicable.

References

- 1 Frank Mittelbach and Michel Goossens. *The L^AT_EX Companion*. Tools and Techniques for Computer Typesetting. Addison-Wesley, Boston, Massachusetts, 2 edition, 2004. With Johannes Braams, David Carlisle, and Chris Rowley. URL: <http://www.informit.com/store/latex-companion-9780201362992>.
- 2 Mark Trettin and Jürgen Fenn. An essential guide to L^AT_EX_{2_ε} usage – obsolete commands and packages, 2007. English Translation of German version. URL: <https://ctan.org/pkg/12tabu-english>.

A Packages used in oasics-v2019

Table 4 lists all packages used in oasics-v2019.

■ **Table 4** Packages used in oasics-v2019.

Package	URL	Selected Options
microtype	https://www.ctan.org/pkg/microtype	
inputenc	https://www.ctan.org/pkg/inputenc	utf8
lmodern	https://www.ctan.org/pkg/lmodern	
fontenc	https://www.ctan.org/pkg/fontenc	T1
textcomp	https://www.ctan.org/pkg/textcomp	
eucal	https://www.ctan.org/pkg/eucal	mathscr
amssymb	https://www.ctan.org/pkg/amssymb	
soul	https://www.ctan.org/pkg/soul	
color	https://www.ctan.org/pkg/color	
xcolor	https://www.ctan.org/pkg/xcolor	usenames
babel	https://www.ctan.org/pkg/babel	
amsmath	https://www.ctan.org/pkg/amsmath	tbtags,fleqn
enumerate	https://www.ctan.org/pkg/enumerate	
graphicx	https://www.ctan.org/pkg/graphicx	
array	https://www.ctan.org/pkg/array	
multirow	https://www.ctan.org/pkg/multirow	
tabularx	https://www.ctan.org/pkg/tabularx	
threeparttable	https://www.ctan.org/pkg/threeparttable	online
listings	https://www.ctan.org/pkg/listings	
lineno	https://www.ctan.org/pkg/lineno	left,mathlines
lastpage	https://www.ctan.org/pkg/lastpage	
hyperref	https://www.ctan.org/pkg/hyperref	
caption	https://www.ctan.org/pkg/caption	labelsep=space, singlelinecheck=false, font=up,small, labelfont=sf,bf, listof=false
rotating	https://www.ctan.org/pkg/rotating	figuresright
subcaption	https://www.ctan.org/pkg/subcaption	
comment	https://www.ctan.org/pkg/comment	
xstring	https://www.ctan.org/pkg/xstring	
amsthm	https://www.ctan.org/pkg/amsthm	
cleveref	https://www.ctan.org/pkg/cleveref	capitalise, noabbrev
aliascnt	https://www.ctan.org/pkg/aliascnt	
thmtools	https://www.ctan.org/pkg/thmtools	

B Changes in oasics-v2019 compared to oasics-v2018

■ New Features

- support of metadata in PDF file (e.g., author, title, keywords)
- revised displaying of author-related funding acknowledgements (now displayed as part of the funding block instead of footnotes)

- added support for `cleveref` package (new document option `cleveref`)
- added support for using `\autoref` for theorem-like environments (new document option `autoref`)
- added new environment `claim` and `claimproof` to realize sub-proofs
- added new environment `proposition`
- added support for double-blind peer review
- explicitly defined/named colors used in style to ease reusing them
- added document option `authorcolumns` to activate displaying author details in two columns
- revised style of algorithm environments provided by `algorithm` or `algorithm2e` packages
- added qed-like symbol to mark end of e.g. definitions (command `\oasicsEnd`)
- Bugfixes
 - fixed problems with theorem-like environments when using `cleveref` and `autoref` (see new features above)
 - switched several URL from `http` to `https`
 - fixed problems with using of ACM 2012 classification (deactivated `subclass` and revised support of `ccsdesc` macro)
 - fixed problem caused by `\\` in title macro
 - fixed problems with using `thm-restate` (added new documentclass option `thm-restate`)
 - fixed bug when using `algorithm2e` package and `cleveref` package
 - fixed problems with `texorpdfstring` in author macro
 - capitalised labels when using `autoref` (e.g. section -> Section)
- Minor changes
 - moved ORCID symbol behind author name³⁵
 - added separator `\and` to split several affiliations
 - added warnings when package `enumitem` or `paralist` are loaded; they manipulate the predefined enumeration styles and are partly incompatible
 - preloaded package `microtype` in style
 - revised spacings/font sizes for top matter

B.1 How to switch from `oasics-v2018` to `oasics-v2019`

The following steps are mandatory to switch a document from the `oasics-v2018` to the `oasics-v2019` document class:

1. Copy the file `oasics-v2019.cls` to your document folder.
2. Open your main \LaTeX file and
 - a. change the document class from `\documentclass[...]{oasics-v2018}` to `\documentclass[...]{oasics-v2019}`;
 - b. remove `\subclass` macro and add `1-n \ccsdesc` macros with appropriate 2012 ACM classifications to the \LaTeX header (e.g., retrieved from https://dl.acm.org/ccs/ccs_flat.cfm); For instance, change


```
\subclass{
  \ccsdesc[100]{General and reference~General literature},
  \ccsdesc[100]{General and reference}
}
```

³⁵ see <https://orcid.org/content/journal-display-guidelines>

to

```
\ccsdesc[100]{General and reference~General literature}
\ccsdesc[100]{General and reference}
```

- c. Optional: Check the typesetting of the funding acknowledgements; in `oasics-v2019`, general funding acknowledgements provided with `\funding` macro and individual funding acknowledgements provided as parameter of the `\author` macro are displayed combined.

B.2 How to switch from `oasics-v2016` to `oasics-v2019`

The following steps are mandatory to switch a document from the `oasics-v2016` to the `oasics-v2019` document class:

1. Copy the files `oasics-v2019.cls` and `orcid.pdf` to your document folder.
2. Open your main `LATEX` file and
 - a. change the document class from `\documentclass[...]{oasics-v2016}` to `\documentclass[...]{oasics-v2019}`;
 - b. for every author, revise the existing `\author` macro by
 - removing the (optional) affiliation number (e.g., change `\author[1]{John Doe}` to `\author{John Doe}`),
 - adding four (empty) fields to the `\author` macro (e.g., change `\author{John Doe}` to `\author{John Doe}{ }{ }{ }`),
 - moving the affiliation (incl. the address) to the second field of the `\author` macro (e.g., `\author{John Doe}{Dummy University, City, Country}{ }{ }`),
 - moving the email address to the third field of the `\author` macro,
 - deleting `\affil` commands,
 - (optionally) adding the ORCID id as forth field of the `\author` macro,
 - c. remove `\subjclass` macro and add 1–*n* `\ccsdesc` macros with appropriate 2012 ACM classifications to the `LATEX` header (e.g., retrieved from <https://submission.dagstuhl.de/services/acm-subject-classification>);

The following steps are optional:

1. Add a reference to a related version of the document (e.g., a reference to an extended or full version mentioned usually as a title footnote or at end of the introduction) to the `\relatedversion` macro (e.g., `\relatedversion{A full version of the paper is available at \url{XXX}.}`)
2. Move funding acknowledgements (e.g., mentioned in an acknowledgements section or in a footnote associated to the title) to the `\funding` macro.
3. Move acknowledgements from a separate section (e.g., provided at the end of the conclusions) to the `\acknowledgements` macro.

C basics-v2019 styles of lists, enumerations, and descriptions

List of different predefined enumeration styles:

■ `\begin{itemize}...\end{itemize}`

■ ...

■ ...

1. `\begin{enumerate}...\end{enumerate}`

2. ...

3. ...

(a) `\begin{alphaenumerate}...\end{alphaenumerate}`

(b) ...

(c) ...

(i) `\begin{romanenumerate}...\end{romanenumerate}`

(ii) ...

(iii) ...

(1) `\begin{bracketenumerate}...\end{bracketenumerate}`

(2) ...

(3) ...

Description 1 `\begin{description} \item[Description 1] ... \end{description}`

Description 2 Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui.

Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

Description 3 ...

D basics-v2019 theorem-like environments

List of different theorem-like environments:

► **Theorem 3.** `\begin{theorem} ... \end{theorem}`

theorem

Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

► **Lemma 4.** `\begin{lemma} ... \end{lemma}`

lemma

Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

► **Corollary 5.** `\begin{corollary} ... \end{corollary}`

corollary

Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

► **Proposition 6.** `\begin{proposition} ... \end{proposition}`

proposition

Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

► **Exercise 7.** `\begin{exercise} ... \end{exercise}`

exercise

Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

definition	<p>► Definition 8. <code>\begin{definition} ... \end{definition}</code></p> <p><i>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</i></p>
example	<p>► Example 9. <code>\begin{example} ... \end{example}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</p>
note	<p>► Note 10. <code>\begin{note} ... \end{note}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</p>
note*	<p>► Note. <code>\begin{note*} ... \end{note*}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</p>
remark	<p>► Remark 11. <code>\begin{remark} ... \end{remark}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</p>
remark*	<p>► Remark. <code>\begin{remark*} ... \end{remark*}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</p>
claim	<p>▷ Claim 12. <code>\begin{claim} ... \end{claim}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</p>
claim*	<p>▷ Claim. <code>\begin{claim*} ... \end{claim*}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.</p>
proof	<p>Proof. <code>\begin{proof} ... \end{proof}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque. ◀</p>
claimproof	<p>Proof. <code>\begin{claimproof} ... \end{claimproof}</code></p> <p>Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque. ◀</p>