

Author's Guide to the ACM Book Series Class
(acm-book.cls)

Association for Computing Machinery

February 19, 2019

<i>Class file author</i>	Paul C. Anagnostopoulos, Windfall Software, paul [at] windfall.com
<i>Guide author</i>	Paul C. Anagnostopoulos
<i>ACM representative</i>	M. Tamer Özsu

The programs and applications presented in this book have been included for their instructional value. They have been tested with care, but are not guaranteed for any particular purpose. The publisher does not offer any warranties or representations, nor does it accept any liabilities with respect to the programs or applications.

Copyright © 2014–2019 by the Association for Computing Machinery

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or any other media embodiments now known or hereafter to become known, without the prior written permission of the publisher.

Contents

1	Introduction	1
1.1	What You Need	1
1.2	Status of the Class File and Author's Guide	1
1.3	Sample Files	2
2	Root File	3
2.1	Example	3
2.2	The Document Class	3
2.3	Packages	4
2.4	Definitions	4
2.5	Index	5
2.6	Text	5
3	Front Matter	7
3.1	Example	7
3.2	Title and Author	7
3.3	Contents	7
3.4	Preface	7
4	Hierarchy	9
4.1	Chapters and Appendixes	9
4.2	Sections	9
5	Basic Elements: Lists, Quotes, Etc.	11
5.1	Lists	11
5.2	Quotations	11
5.3	Program Code	11
5.4	Sidebars	12
6	Preparing Art	13
6.1	Required Format	13
6.2	Images	13
6.3	Applications	13

iv CONTENTS

7	Bibliography	15
7.1	What You Need to Provide	15
7.2	Guidelines	16
7.2.1	Uniformity	16
7.2.2	Standard Database Fields	16
8	Index	21
8.1	Index Style	21
9	Style	23
9.1	Punctuation	23
9.2	Grammar	23
9.3	URLs	24
	Sample Bibliography	25

1 Introduction

The Association for Computing Machinery book series style is a \LaTeX class file that you use to prepare books for their Association for Computing Machinery series. The class file is named `acm-book.cls`. It produces a manuscript that looks roughly like finished books in the series. Please note that your manuscript will not look exactly like the finished book.

The ACM class file is a variant of the standard \LaTeX book style. It is based on `book.cls` and both replaces and adds to its features. This author's guide assumes you are familiar with \LaTeX and describes the features of the ACM class file that are new or different.

The ACM style was commissioned by Morgan & Claypool Publishers and implemented by Paul C. Anagnostopoulos, with the assistance of Tamer Ozsu. Please address questions and problems to Paul C. Anagnostopoulos, Windfall Software, paul [at] windfall.com.

1.1 What You Need

You need the following files to use the ACM class file:

- This document, `acm-guide.pdf`
- The \LaTeX class file, `acm-book.cls`
- The bibliography database file with standard abbreviations, `acm-publications.bib`. See Section 7.2.1.
- The bibliography style file, `acm-book.bst`. See Chapter 7.
- The set of sample files that give examples of the class usage and act as templates for your book files.

The latest versions of these files are available at <http://books.acm.org/authors/author-instruction>

1.2 Status of the Class File and Author's Guide

Version 1.0 of the class file and the matching author's guide were released in Spring 2014. The class file contains a complete revision history at the end, beginning with its creation in February 2014.

The date on the title page of this guide can be used with the revision history to determine which version of the class file this guide corresponds to.

1.3 Sample Files

The following sample files accompany the `acm-book.cls` file. Please note that the text in the files does not represent an actual book in the ACM series.

acm-book.tex This is a template root file that acts as the main file for your book. This is the file that you process with LaTeX. It contains the `\documentclass` command along with an `\include` command for each division of your book. See Chapter 2.

It is important to use a root file with `\include` commands so that your book is processed as a coherent whole, rather than simply treating each chapter as a separate book.

front.tex This is a template for the front matter of your book. It shows how to code the book title, your name and affiliation, the table of contents, and the preface.

preface.tex This is a sample preface, input as part of the front matter.

chap01.tex This is a sample chapter file. It contains examples of the chapter title, section heads, subsection heads, figures, a simple table, and an exercise list.

appa.tex This is a sample appendix file.

bio.tex This is a sample author biography file.

acm-publications.bib This BiBTeX database includes standard abbreviation for journal names. See Section 7.2.1.

2 Root File

This chapter describes the organization of your book's root file, including its prolog and text. The prolog is the portion of the \LaTeX root file that precedes the `\begin{document}` command.

2.1 Example

```
\documentclass[times]{acm-book}

\usepackage{amsmath}    % Package for AMS math extensions.
\usepackage{graphicx}    % Package for importing EPS files.
\usepackage{mybook}      % .sty file with your custom macros.

\makeindex               % if you use \index{...} in your book.

\begin{document}

\frontmatter
\include{front}          % Front matter.

\mainmatter
\include{chap01}          % Chapter 1.
\include{chap02}
\include{chap03}
\include{appa}           % Appendix A.

\backmatter
\bibliography{acm-publications,biblio} % Bibliography
\include{bio}             % Author(s) biographies.
\include{book.ind}        % If the book has an index.

\end{document}
```

2.2 The Document Class

The `\documentclass` command names the ACM class file and lists any desired options.

Table 2.1 Document class options

Option	Default?	Description
<code>cm</code> , <code>computermodern</code>	yes	Use Computer Modern fonts.
<code>mathtime</code>		Use the MathTime math fonts. This is recommended when using the Times Roman fonts.
<code>times</code>		Use Times Roman fonts.

Table 2.2 Packages used by the ACM class

Package	Used
<code>mathtime</code>	when the ‘ <code>mathtime</code> ’ option is specified
<code>natbib</code>	the <code>natbib</code> package is the standard for bibliographies
<code>url</code>	provides <code>\url</code> command for specifying URLs

▷ `\documentclass[option-list]{acm-book}`

The *option-list* argument is optional, and so is enclosed in square brackets if specified. Table 2.1 lists the available options and notes which ones are present by default.

2.3 Packages

If you need to use any \LaTeX packages, these are specified immediately following the `\documentclass` command. Table 2.2 lists the packages that are used by the ACM class.

▷ `\usepackage{amsmath}`

2.4 Definitions

If you need to define any macros or other \LaTeX entities for use in writing the book, it is best to collect them in a separate file whose name is the same as the root file and whose extension is `.sty`. This file is treated as a package and loaded with a `\usepackage` command after the other \LaTeX packages you are using.

▷ `\usepackage{mybook}`

2.5 Index

We want each of our books to have an index. Using \LaTeX this is easy – simply place `\index{}` commands in the text next to the terms you want to be indexed. For example: indexing this phrase `\index{this phrase}`. To produce the index, you must specify the following command in the preamble:

▷ `\makeindex`

2.6 Text

The text of your book appears between the `\begin{document}` and `\end{document}` commands. Instead of placing the actual text in the root file, it is much better to distribute it to multiple files that are then named in the root file with `\include` commands. We recommend that you create a separate file for each of these divisions:

- Root file
- Front matter
 - Preface
- Each chapter
- Each appendix
- Bibliography (`.bib` and `.bbl` files)
- Author biographies
- Index

3 Front Matter

This chapter describes the information you need to provide in the front matter of your book. This is the portion of the book that precedes the first chapter. We recommend that you put all this information in a file named `front.tex`.

Do not spend a lot of time formatting your front matter. This is a job for the compositor, the person who formats the text and lays out the pages.

3.1 Example

```
\title{Title of Your Book}

\author{Your Name}
\thanks{Department of Computer Science\\
Brown University\\
Providence, RI}

\maketitle

\tableofcontents

\input{preface}
```

3.2 Title and Author

The portion of the front matter preceding the `\maketitle` command specifies the title of your book, your name, and your affiliation. The `\thanks` command is co-opted to specify the affiliation.

3.3 Contents

The `\tableofcontents` command produces the table of contents.

3.4 Preface

The final command in the front matter file includes the separate preface file, `preface.tex`, in the front matter. The preface should begin with a quasi-chapter command to start it on a new page:

8 Chapter 3 *Front Matter*

▷ `\chapter*{Preface}`

Many authors like to include acknowledgments in their prefaces. This can be the final section of the preface, or, if it is particularly large, a separate quasi-chapter in a separate file named `acks.tex`.

4 Hierarchy

This chapter describes the commands that are used to produce hierarchical headings in your paper.

4.1 Chapters and Appendixes

- ▷ `\chapter{title}`

This command produces a numbered chapter heading, beginning on a new page.

- ▷ `\chapter*{title}`

This variant produces an unnumbered “quasi-chapter” heading, beginning on a new page. Quasi-chapters include the preface, bibliography, author biographies, and index.

- ▷ `\appendix`

This command precedes a `\chapter` command and flags it as the *first* appendix in the book. It should appear on a line by itself immediately preceding the `\chapter` command.

- ▷ `\chapterauthor{name(s)}`

This command can be used in an edited volume with multiple authors to specify the authors for a particular chapter or appendix. It should be positioned immediately following the `\chapter` command.

4.2 Sections

- ▷ `\section{title}`

This command produces a numbered section heading. The heading is set off from the next paragraph, which is not indented. Please use numbered sections for all sections in your book, except for those listed next.

- ▷ `\section*{title}`

This variant produces an unnumbered section heading. These should be used only for the following kinds of sections: Exercises, Problems, Solutions, References.

- ▷ `\subsection{title}`

This command produces a numbered subsection heading that is numbered within the preceding section. The heading is set off from the next paragraph, which is

not indented. Please use numbered subsections for all subsections that are within numbered sections.

▷ `\subsection*{title}`

This variant produces an unnumbered subsection heading.

▷ `\subsubsection{title}`

This command produces a numbered sub-subsection heading that is numbered within the preceding subsection. The heading is set off from the next paragraph, which is not indented.

▷ `\subsubsection*{title}`

This variant produces an unnumbered sub-subsection heading.

▷ `\paragraph*{title}`

This command produces an unnumbered level-4 heading that is run in with the first line of text.

5 Basic Elements: Lists, Quotations, Etc.

This chapter lists the basic L^AT_EX elements that are available in the class file.

5.1 Lists

The class file provides four levels of lists for each of the standard list types: bulleted, numbered, and labeled. The bullet items at each level are preceded by a large round bullet, small square, dash, and small bullet. The numbered items at each level are preceded by an arabic number, lowercase letter, roman numeral, and uppercase letter.

5.2 Quotations

The class file provides the `quote` environment for short quotations and the `quotation` environment for multiparagraph quotations.

5.3 Program Code

The `verbatim` environment can't be used for displaying program code in a monospaced font. The text in the environment is taken verbatim, with no interpretation of the usual L^AT_EX commands or special characters.

```
if directives? then
  node.nlexeme := format("rcs_replacef(\"~S\"", fcs);
  foreach i in reverse vector fcs_arg do
    node.nlexeme &:= "," & fcs_arg[i];
  end
  node.nlexeme &:= ")";
else
  node.nlexeme := format("replace(\"~S\")", fcs);
fi
```

If you need a program code environment where L^AT_EX commands can still be used, consider the `alltt` package, which is available at <http://www.ctan.org/pkg/alltt>. A typical use of L^AT_EX commands in program code is to indicate metasyntactic variables in *italic*.

5.4 Sidebars

The class file provides the `sidebar` environment for producing sidebars in your book. A sidebar is a floating element, like figures and tables, and so is positioned at the top or bottom of pages. It is numbered and has a title. It typically contains text, lists, and perhaps small figures. The purpose of a sidebar is to present ancillary material that can be read out of order or sometimes even skipped.

```
▷ \begin{sidebar}{title}
▷ ...
▷ \end{sidebar}
```

We recommend that each sidebar be referenced at the appropriate point in the text, to give the reader a clue when to read it.

6 Preparing Art

Preparing art files for your book can be one of the most complex and time-consuming aspects of writing. In this chapter we will present some guidelines to help you produce better art files. Note, however, that in many cases your art will have to be redrafted by a professional artist in order to be of the required format and quality. In particular, the font families used in the published ACM series books are not the same as those used by the ACM book series class.

6.1 Required Format

The standard art file format for published books is Encapsulated PostScript (EPS). Some compositors (typesetters) preview book pages with an application that does not include a PostScript interpreter, so a bitmapped preview is required in each EPS file. The preview should be an 8-bit color tiff bitmap.

If you cannot produce EPS files with 8-bit tiff previews, the next best thing are EPS files without previews. It is relatively easy for the compositor to add the previews.

If you cannot produce EPS files at all, then the next choice is PDF files. Some compositors can place PDF files directly on the book pages, others will need to export EPS files using Adobe Acrobat.

6.2 Images

A bitmapped image must be at least 240 DPI to look reasonable on a printed page. If you simply grab an image from the Web, for example, it will rarely be more than 100 DPI. Also note that there are permission issues with copyrighted material on the Web.

If you are generating images with standard or home-grown applications, try to adjust the resolution to 240 DPI or greater. If an application generates 72-DPI images and nothing else, then be sure to make the image at least three times as large as required, and preferably four times as large.

6.3 Applications

The industry-standard applications for creating professional art are Adobe Illustrator and PhotoShop, and Corel CorelDraw. Unfortunately, these applications are expensive and difficult to learn. If you use another standard or home-grown application, please try to follow the guidelines in the previous section.

7 Bibliography

This chapter describe the bibliography facilities provided by the ACM package.

The ACM class automatically loads the `natbib` package and enables author/year citation mode using its ‘`authoryear`’ option. Table 7.1 shows the various citation commands that are available and what they produce. The class also establishes the `acm-book` bibliography style, which was developed specifically for the ACM book series.

Try to construct sentences with citations so that if the bracketed citation is removed, the sentence still makes sense. For example,

As described in [Smith et al. 1991], the database contains...

is better written as

As Smith et al. [1991] describe, the database contains...

If the citation is incidental and does not act as a noun in the sentence, then put it at the end:

The database contains... [Smith et al. 1991].

7.1 What You Need to Provide

You can produce your bibliography using BibTeX or any other facility that ultimately generates a `.bbl` file. When you submit your book files to the ACM, submit the final `.bbl` file. We typeset your bibliography from this file, not from the bibliography

Table 7.1 Bibliography citation commands

Command	Produces
<code>\citet{label}</code>	Smith et al. [1991]
<code>\citet{label1,label2}</code>	Smith et al. [1991], Balzer [2007]
<code>\citep{label}</code>	[Smith et al. 1991]
<code>\citep{label1,label2}</code>	[Smith et al. 1991, Balzer 2007]
<code>\citep[p. 123]{label}</code>	[Smith et al. 1991, p. 123]
<code>\cite</code>	same as <code>\citep</code>

database or any other “source” files. Please be sure to generate the final `.bbl` file using the `acm-book.bst` bibliography style file.

7.2 Guidelines

7.2.1 Uniformity

One of the major causes of concern is the non-uniform way in which authors use journal and conference names. Sometimes authors use *ACM Transactions on Database Systems* and other times *ACM TODS*. The variation on conference titles is even worse. In some cases we have seen non-uniform use of journal/conference names even within the same work. We would like to see common and uniform expansions. Therefore,

- We provide the `acm-publications.bib` file that includes abbreviations and their expansions. Please put this in the directory where you keep your `.bib` files and include it in the root file `\bibliography` command as follows:

▷ `\bibliography{publications,bibfile...}`

The order of file names is important.

- The `publications` file consists of entries of the form:

▷ `@STRING{abbreviation = {Expanded title}}`

where *abbreviation* is the abbreviation in the `.bib` file (see below) that will expand to *Expanded title* when processed with \LaTeX and \BibTeX .

- In the `.bib` entries, use the `abbreviation` in `booktitle` and `journal` entries (without any quotes surrounding the abbreviation) and \BibTeX will do the rest.
- If you come across a journal or conference that you wish to reference that is not in the `publications.bib` file, please add it to the end of the `publications.bib` file. Include this file in the files you send back and we will update the master copy.

7.2.2 Standard Database Fields

Please use standard database fields for references. Here is what we use for the various kinds of entries.

@article For articles, use at least the following fields: `author`, `title`, `journal`, `volume`, `number`, `pages`, `year`. In some journals, there is no issue number, in which case this can be omitted. Some journals (e.g., ACM journals) have recently done away with page numbers, but instead use paper number in an issue or volume. In this case, please use `Paper n` as the value of the `pages` field. An example of fully formed journal entry is Straube and Özsu [1990], which is produced by the following (note the abbreviation `TOIS` used in the `journal` field):

```
@article{STR90a,
  author = {D.D. Straube and M. T. {"0}zsu},
  title = {Queries and Query Processing in Object-Oriented
    Database Systems},
  journal = TOIS,
  volume = {8},
  number = {4},
  pages = {387--430},
  year = 1990}
```

@book For books, use at least the following fields: **author**, **title**, **publisher**, **address**, **year**. An example is Levin [2002], which is produced by the following database entry:

```
@book{Levin2002,
  author = {Levin, F. S.},
  publisher = {Cambridge University Press},
  address = {Cambridge, England},
  title = {An Introduction to Quantum Theory},
  year = {2002}}
```

In the case of an edited book, please use the **editor** field instead of **author**.

@inbook For chapters that appear in a book, use at least the following fields: **author**, **title**, **pages**, **publisher**, **address**, **year**. An example is Tygar and Yee [1991], produced by:

```
@inbook{yee91strongbox,
  author = {J. D. Tygar and Bennet S. Yee},
  title = {Strongbox: A System for Self-Securing Programs},
  publisher = {Addison-Wesley},
  address = {Reading, MA},
  chapter = {7},
  pages = {234--278},
  year = 1991
}
```

@incollection For papers that appear in collections, use at least the following fields: **author**, **title**, **booktitle**, **editor**, **pages**, **publisher**, **year**. An example is Özsu and Yao [2001], produced by the following:

```
@incollection{OzsuYao01,
```

```

author = {\0}zsu, M. Tamer and Yao, Bin},
title = {Building Component Database Systems Using {CORBA}},
booktitle = {Component Database Systems},
editor = {Dittrich, Klaus and Geppert, Andreas},
pages = {207--236},
publisher = MORGAN,
year = {2001}}

```

@inproceedings For conference papers, use at least: **author**, **title**, **booktitle**, **pages**, **year**. An example is Zhang et al. [2008], produced by the following:

```

@inproceedings{WangD008,
  author = {Zhang, Q. and Daudjee, K. and {\0}zsu, M. T.},
  title = {Popularity-aware Prefetch in {P2P} Range Caching},
  booktitle = P2P08,
  pages = {53--62},
  year = 2008}

```

Of course, if you are citing the original book (as in Doğaç et al. [1994]), you can use the **crossref** field instead of entering all of the info; see Özsu [1994]. These are produced by the following:

```

@book{natoasi93,
  editor = {A. Do{\v{g}}a{\c{c}} and M.T. {\0}zsu and A. Biliris
           and T. Sellis},
  publisher = {Springer},
  address = {Philadelphia, PA},
  title = NATO93,
  year = {1994}}

```

```

@incollection{ozsu94,
  author = {\0}zsu, M. T.},
  crossref = {natoasi93},
  pages = {147--184},
  title = {Transaction Models and Transaction Management in
           Object-Oriented Database Management Systems}}

```

This also applies to the **@inbook** type.

`@mastersthesis`, `@phdthesis`, `@techreport` For theses and reports, use at least: `author`, `title`, `school`, `address`, `year`. An example is Chou [1985], produced by the following:

```
@phdthesis{Chou85,
  author = {Chou, H.},
  title = {Buffer Management of Database Systems},
  school = {Department of Computer Science, University of Wisconsin},
  address = {Madison, WI},
  year = {1985}}
```

`@website` For a Web site, use at least: `author`, `title`, `year`, `url`, `retrieved`. And example is Anagnostopoulos [2015], produced by the following:

```
@website{windfall,
  author = {Paul C. Anagnostopoulos},
  title = {Windfall Software Web site},
  year = {2015},
  url = {http://www.windfall.com},
  retrieved = {February 2016}}
```


8 Index

All books in the ACM book series have indexes. The ACM appreciates it if you prepare your own index using the L^AT_EX indexing facilities. The index is copyedited along with the rest of the book and, if needed, you will have the opportunity to expand the index when reviewing page proof.

8.1 Index Style

Please follow these style guidelines when creating your index.

- Main headings should have the first word capitalized. The remaining words should be lowercase unless they are proper names (e.g., “Coherence protocol,” “Neuromorphic chips,” “AMD Opteron”).
- Subheadings and sub-subheadings should be lowercase unless they are proper names.
- Full names should be in the format *Lastname, Firstname* (e.g., Knuth, Donald).
- When the index includes an acronym and its expansion, the page information should be included under whichever is the most commonly used in the book. If the acronym is commonly used, list the information under the acronym. If the expansion is commonly used, list the information under the expansion. Then index the other item with a “See” reference to the item that has the information.
- If you find that a main heading is accumulating more than about five page numbers, consider adding subheadings that give the reader better hints about which pages they want to read.
- Pay attention to indexing the concepts in your book. Indexes for technical books tend to include specific examples of concepts without also including the pages that introduce, define, and elaborate the concepts.
- Index important terms, including those not present in the chapter, section, and subsection headings.
- Avoid listing every mention of proper nouns (these are often picked up in automated subject searches). Index only subjects discussed in your book and not those that are just mentioned in passing.

- If you think a reader might try to look up a term, index it. There is no harm in having more entries in the index.
- On average, one or two index terms per book page is included in the index.
- Do *not* index the Table of Contents, Preface (unless it contains information not discussed elsewhere), References/Bibliography, Glossary, or Biographies.

9 Style

This chapter provides information on writing style, both requirements and suggestions.

9.1 Punctuation

- Please use the “serial” or “Oxford” comma in lists. That is, every item except the first should be preceded by a comma (or a semicolon if the items are complex). For example:

It is necessary to provide names for symbols, classes, packages, and databases.

We begin by describing the use of the semicolon; the need for hyphens, en dashes, and dashes; and the proper use of parentheses.

- Hyphens are the short dash used in hyphenated words, coded as a single hyphen (-). A common error is to use an en dash for a hyphen; please avoid this.
- Number ranges are punctuated with the “en dash,” which is coded in L^AT_EX as two consecutive hyphens (--). One common use of a number range is for page ranges in bibliographic entries.

In the years 1964–1970, many new computer architectures...

- The long dash used for punctuation is called an “em dash” and is coded as three consecutive hyphens (---). The dash indicates an interruption or abrupt change of thought, as in:

You are the person—the only friendly person—who offered to help with this issue.

It is best to use the dash sparingly in technical writing. Also, do not use a dash where a colon better serves to introduce material.

9.2 Grammar

- In computer science, we use the term *data* quite frequently. There is a debate about its proper use, centered around whether the noun is singular (*data* is) or plural (*data* are). Strictly speaking, *data* is the plural of *datum*; however, over time its everyday use has shifted and it is now considered a mass noun that can take a singular verb. We don’t have a rule about its usage and will accept both; however, we ask you to be consistent in your usage.

- *that* versus *which*. The word *that* is used to set off a restrictive clause; that is, a clause that restricts the identity of the subject. For example:

The car that was parked in the driveway was stolen.

The word *which* is used to set off a nonrestrictive clause; that is, a clause that specifies something interesting but incidental about the subject. For example:

The car, which I bought only last week, was stolen right out of my driveway.

Restrictive clauses (*that*) are not preceded by a comma; nonrestrictive clauses (*which*) are preceded by a comma.

9.3 URLs

When you include a URL, please specify it fully, including the protocol: `http://www.windfall.com`. Also, please add the phrase “(Last retrieved: *date*)” to indicate that the link was operational on the date that you specify.

A URL should be typeset using the `\url{...}` command, which takes care of special characters in the URL (periods, ampersands, etc.). It also creates a clickable hyperlink to the site in the final PDF file. The `url` package is automatically used by the class file

As an example,

```
\url{http://db.uwaterloo.ca/~tozsu/}
```

produces the URL `http://db.uwaterloo.ca/~tozsu/`.

Bibliography

- P. C. Anagnostopoulos. 2015. Windfall software web site. Available at <http://www.windfall.com>; last retrieved January 2019.
- H. Chou. 1985. *Buffer Management of Database Systems*. PhD thesis, Department of Computer Science, University of Wisconsin, Madison, WI.
- A. Doğaç, M. T. Özsu, A. Biliris, and T. Sellis, eds. 1994. *Advances in Object-Oriented Database Systems*. Springer. Philadelphia, PA.
- F. S. Levin. 2002. *An Introduction to Quantum Theory*. Cambridge University Press, Cambridge, England.
- M. T. Özsu. 1994. Transaction models and transaction management in object-oriented database management systems. In Doğaç et al. [1994], pp. 147–184.
- M. T. Özsu and B. Yao. 2001. Building component database systems using CORBA. In K. Dittrich and A. Geppert, eds., *Component Database Systems*, pp. 207–236. Morgan-Kaufmann.
- D. D. Straube and M. T. Özsu. 1990. Queries and query processing in object-oriented database systems. *ACM Trans. Information Syst.*, 8(4): 387–430.
- J. D. Tygar and B. S. Yee. 1991. *Strongbox: A System for Self-Securing Programs*, chapter 7, pp. 234–278. Addison-Wesley, Reading, MA.
- Q. Zhang, K. Daudjee, and M. T. Özsu. 2008. Popularity-aware prefetch in P2P range caching. In *Proc. 8th IEEE Int. Conf. on Peer-to-Peer Computing*, pp. 53–62.