This bibliography may be freely used for non-commercial purposes. It may also be freely distributed provided that this notice is included. I would be most grateful to receive additions, corrections and URLs of electronically available papers. The bibliography is also available in BibTeX and HTML forms from https://www.cs.kent.ac.uk/people/staff/rej/gcbib/gcbib.html

Copyright © 1999-2019, Richard Jones


[Amsaleg et al., 1995b] Laurent Amsaleg, Michael Franklin, and Olivier Gruber. Efficient incremental garbage collection for client–server object database systems. In Twenty-first International Conference on Very Large Databases (VLDB95), Zurich, Switzerland, September 1995.


[Andreasson et al., 2002] Eva Andreasson, Frank Hoffmann, and Olof Lindholm. To collect or not to collect? machine learning for memory management. In JVM 2002 [JVM 20022002].


[Bozman, 1984] Gerald Bozman. The software lookaside buffer reduces search overhead with linked

[Braberman et al., 2008] Víctor Braberman, Federico Fernández, Diego Garbervetsky, and Sergio
Yovine. Parametric prediction of heap memory requirements. In Jones and Blackburn
[Jones and Blackburn2008], pages 141–150.

[Branquart and Lewi, 1971] P. Branquart and J. Lewi. A scheme of storage allocation and garbage col-

[Brecht et al., 2001] Tim Brecht, Eshrat Arjomandi, Chang Li, and Hang Pham. Controlling garbage
collection and heap growth to reduce the execution time of Java applications. In OOPSLA 2001
[OOPSLA 20012001].

[Brecht et al., 2006] Tim Brecht, Eshrat Arjomandi, Chang Li, and Hang Pham. Controlling garbage
collection and heap growth to reduce the execution time of Java applications. *ACM Transactions on
Programming Languages and Systems*, 28(5), September 2006.

[Brega and Rivera, 2000] Roberto Brega and Gabrio Rivera. Dynamic memory management with
garbage collection for embedded applications. In *Proc. of the USENIX Workshop on Industrial Expe-


[Broberg et al., ] Magnus Broberg, Daniel Häggander, Per Lidén, and Lars Lundberg. Improving the

collection for storage-oriented clusters. In V. Estivill-Castro, editor, *Proceedings of Australian Com-


performance and portability. In *International Joint Conference on Artificial Intelligence*, volume 2,

[Brooks, 1984] Rodney A. Brooks. Trading data space for reduced time and code space in real-time
garbage collection on stock hardware. In Steele [Steele1984], pages 256–262.

[Brooksby, 2002] Richard Brooksby. The Memory Pool System: Thirty person-years of memory man-

University of Newcastle upon Tyne, September 1984.

Jouannaud [Jouannaud1985].

[Broy and Pepper, 1982] Manfred Broy and Peter Pepper. Combining algebraic and algorithmic reason-
ing: An approach to the Schorr–Waite algorithm. *ACM Transactions on Programming Languages and

[Bruha, 1987] Ivan Bruha. Representation of structures and garbage collection in McMaster POPLOG.
Technical Report 88-01, McMaster University, Department of Computer Science and Systems,Canada,
1987.


22


Chang and Kuo, 2002] Li-Pin Chang and Tei-Wei Kuo. A real-time garbage collection mechanism for flash-memory storage systems in embedded systems. In RTCSA 2002 [RTCSA 20022002].


[Click et al., 2005] Cliff Click, Gil Tene, and Michael Wolf. The Pauseless GC algorithm. In Hind and Vitek [Hind and Vitek2005], pages 46–56.


[Curial et al., 2008] Stephen Curial, Peng Zhao, Jose Nelson Amaral, Yaoqing Gao, Shimin Cui, Raul Silvera, and Roch Archambault. Memory pooling assisted data splitting (MPADS). In Jones and Blackburn [Jones and Blackburn2008], pages 101–110.


[Dillig et al., 2008] Isil Dillig, Thomas Dillig, Eran Yahav, and Satish Chandra. The CLOSER: Automating resource management in Java. In Jones and Blackburn [Jones and Blackburn2008], pages 1–10.


35


[Edwards, Date unknown] Daniel J. Edwards. Lisp II garbage collector. AI Memo 19, MIT AI Laboratory, Date unknown.


[Fink and Qian, 2003] Stephen J. Fink and Feng Qian. Design, implementation and evaluation of adaptive recompilation with on-stack replacement. In CGO 2003 [CGO 20032003], pages 241–252.


[Hicks, 1993] James Hicks. Experiences with compiler-directed storage reclamation. In Hughes [Hughes1993].


52


54


[Jones and Ryder, 2008] Richard Jones and Chris Ryder. A study of Java object demographics. In Jones and Blackburn [Jones and Blackburn2008], pages 121–130.


Jung and Yi, 2008] Yungbum Jung and Kwangkeun Yi. Practical memory leak detector based on parameterized procedural summaries. In Jones and Blackburn [Jones and Blackburn2008], pages 131–140.


[Kurihara et al., 1990] Satoshi Kurihara, Mikio Inari, Norihisa Doi, Kazuki Yasumatsu, and Takemi Yamazaki. SPICE collector : The run-time garbage collector for Smalltalk-80 programs translated into C. In Jull and Juul [Jul and Juul1990].


70


[Moss et al., 1993] Eliot Moss, Paul R. Wilson, and Benjamin Zorn, editors. OOPSLA Workshop on Garbage Collection in Object-Oriented Systems, October 1993.


83


[Phan et al., 2008] Quan Phan, Gerda Janssens, and Zoltan Somogyi. Runtime support for region-based memory management in Mercury. In Jones and Blackburn [Jones and Blackburn2008], pages 61–70.


[Piumarta et al., 1995] Ian Piumarta, Marc Shapiro, and Paulo Ferreira. Garbage collection in distributed object systems. In Workshop on Reliability and Scalability in Distributed Object Systems, OOPSLA’95, Austin, TX, October 1995.


[Pizlo et al., 2008a] Filip Pizlo, Erez Petrank, and Bjarne Steensgaard. Path specialization: Reducing phased execution overheads. In Jones and Blackburn [Jones and Blackburn2008], pages 81–90.


[Plainfosee and Shapiro, 1992] David Plainfosee and Marc Shapiro. A distributed GC in an object-support operating system. In Cabrera et al. [Cabrera et al.1992].


96


Puaut, 1994a Isabelle Puaut. A distributed garbage collector for active objects. In PARLE’94 [PARLE94], Also INRIA UCIS-DIFUSION RR 2134.


[Richer and Shapiro, 2001] Nicolas Richer and Marc Shapiro. The memory behaviour of the WWW, or the WWW considered as a persistent store. In Kirby et al. [Kirby et al.2001], pages 136–146.


[Sartor et al., 2008a] Jennifer B. Sartor, Martin Hirzel, and Kathryn S. McKinley. No bit left behind: Limits of heap data compression. In Jones and Blackburn [Jones and Blackburn2008], pages 111–120.


[Shapiro et al., 1994] Marc Shapiro, David Plainfossé, Paulo Ferreira, and Laurent Amsaleg. Some key issues in the design of distributed garbage collection and references. In Unifying Theory and Practice in Distributed Systems, Dagstuhl (Germany), September 1994.


[Shen and Martonosi, 2006] John Paul Shen and Margaret Martonosi, editors. Proceedings of the Twelfth International Conference on Architectural Support for Programming Languages and Operating Systems, ACM SIGPLAN Notices 41(11), San Jose, CA, USA, October 2006.


[Shuf et al., 2002b] Yefim Shuf, Manish Gupta, Hubertus Franke, Andrew Appel, and Jaswinder Pal Singh. Creating and preserving locality of Java applications at allocation and garbage collection times. In *OOPSLA 2002* [OOPSLA 20022002].


[Soman et al., 2008] Sunil Soman, Chandra Krintz, and Laurent Daynes. MFM\textsuperscript{2}: Scalable memory management for multi-tasking managed runtime environments. In ECOOP 2008 [ECOOP 20082008].


[SPIN, ] The SPIN operating system. A collection of papers available on the WWW.


[Spoonhower et al., 2006] Daniel Spoonhower, Joshua Auerbach, David F. Bacon, Perry Cheng, and David Grove. Eventrons: A safe programming construct for high-frequency hard real-time applications. In Schwartzbach and Ball [Schwartzbach and Ball2006], pages 283–294.


[Tel and Mattern, 1991] Gerard Tel and Friedmann Mattern. The derivation of distributed termination detection algorithms from garbage collection schemes — (extended abstract). In Aarts et al. [Aarts and others1991].


