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 © 1999-2018, Richard Jones


2


[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].


[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].


Cahill et al., 1993] Vinny Cahill, Seán Baker, Chris Horn, and Gradimir Stavovic. The Amadeus
GRT — generic runtime support for distributed persistent programming. In OOPSLA 1993


Calcagno et al., 2003] Cristiano Calcagno, Peter O’Hearn, and Richard Bornat. Program logic and

2001 [SPACE 2001/2001].

Calder et al., 1994] Brad Calder, Dirk Grunwald, and Benjamin Zorn. Quantifying behavioral differ-

Calder et al., 1998] Brad Calder, Chandra Krintz, S. John, and T. Austin. Cache-conscious data place-


Campos and Hanson, 1993] Alvaro E. Campos and David R. Hanson. Garbage collection in EZ. In
R. Baerz-Yates, editor, Proceedings of Thirteenth International Conference on Computer Science,

PhD dissertation, Princeton University, Department of Computer Science, June 1993.

Cann and Oldehoeft, 1988] D. C. Cann and Rod R. Oldehoeft. Reference count and copy elimination
for parallel applicative computing. Technical Report CS–88–129, Department of Computer Science,


Cardelli et al., 1988] Luca Cardelli, James Donahue, Lucille Glassman, Mick Jordan, Bill Kalsow,
and Olivetti Research Center, 1988.

Cardelli et al., 1992] Luca Cardelli, James Donahue, Lucille Glassman, Mick Jordan, Bill Kalsow,

217.


Dr. Dobb’s Journal, 17(11), November 1992.

Carlsson et al., 1990] Svante Carlsson, Christer Mattsson, and Mats Bengtsson. A fast expected-time
compacting garbage collection algorithm. In Jul and Juul [Jul and Juul1990].


Carlsson, 1987] M. Carlsson. Freeze, indexing and other implementation issues in the WAM. In Lassez
[Lassez1987], pages 40–58.


Cost of garbage collection in the SELF system. In Moss et al. [Moss et al.1993].


[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 2002]*.


24


[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 Blackburn 2008], pages 101–110.


[Dickman and Wilson, 1997] Peter Dickman and Paul R. Wilson, editors. OOPSLA Workshop on Garbage Collection and Memory Management, October 1997.

[Dickman, 1991] Peter Dickman. Effective load balancing in a distributed object-support operating system. In Cabrera et al. [Cabrera et al.1991].


[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.


34


[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].


53


Lorenz Huelsbergen and James R. Larus. A concurrent copying garbage collector for languages that distinguish (im)mutable data. In PPOPP 1993 [PPOPP 19931993], pages 73–82.

Lorenz Huelsbergen and Phil Winterbottom. Very concurrent mark-&-sweep garbage collection without fine-grain synchronization. In Peyton Jones and Jones [Peyton Jones and Jones1998], pages 166–175.


59


[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 Jul and Juul [Jul and Juul1990].


69


74


[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.


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


94


96


[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.


[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.


[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 others 1991].


115


