

# Short-term Memory for Self-collecting Mutators: Benchmarks

Seminar Embedded Software Engineering, Winter 2009/2010

Andreas Haas  
Andreas Schönegger

Universität Salzburg

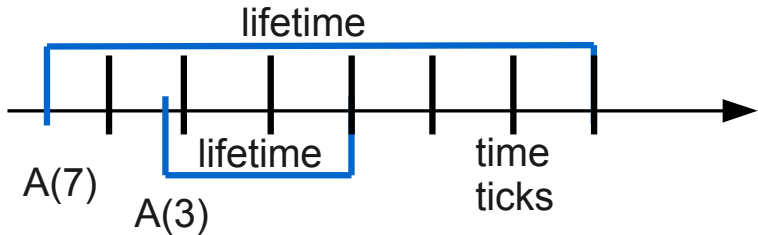
January 28, 2010

- Bestehendes Speichermodell
  - Allozieren - Deallozieren
  - Beispiele: malloc() - free(), Garbage Collector
- Probleme der bestehenden Varianten
- Unsere Vorschlag: “Short-term Memory Model”

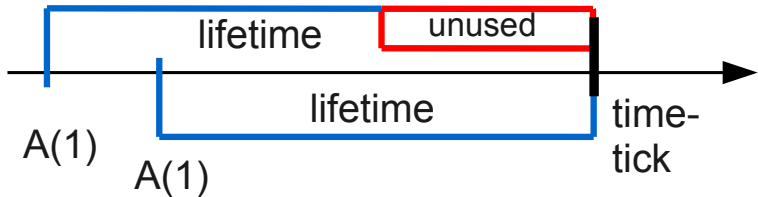
# Short-term Memory Model

- Objectallozierung nur auf Zeit
  - keine Deallozierung notwendig
- Refresh zum Verlängern

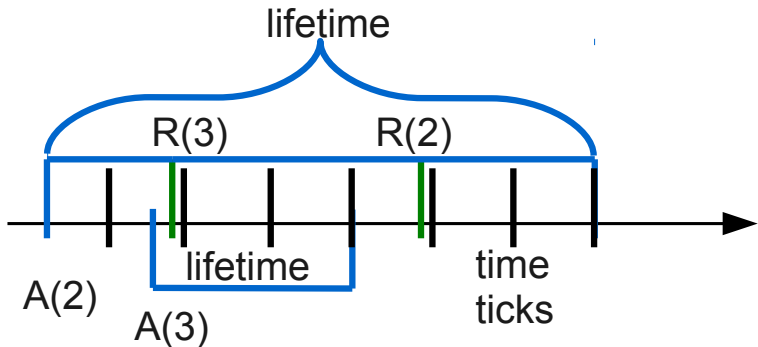
# Grafiken Short-term Memory Model



# Grafiken Short-term Memory Model

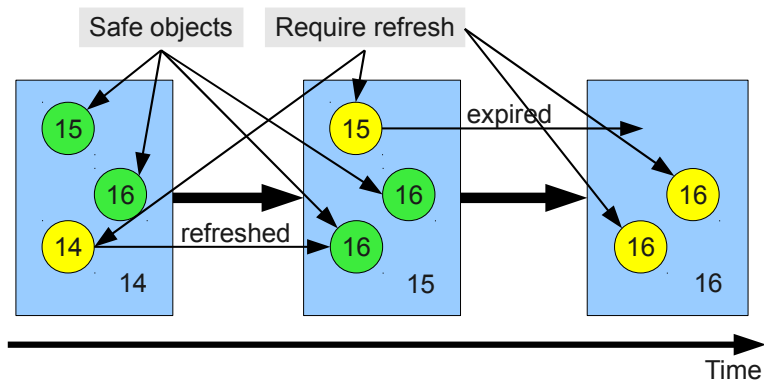


# Grafiken Short-term Memory Model



- Alte Objekte werden überschrieben
  - nur von Objekten der selben Allocation Site
- Zeit

# Tick





- Allokieren =  $O(1)$
- Deallokieren = nicht nötig
- Dereferenzieren =  $O(1)$
- Refresh =  $O(1)$

# Günstige Programstruktur

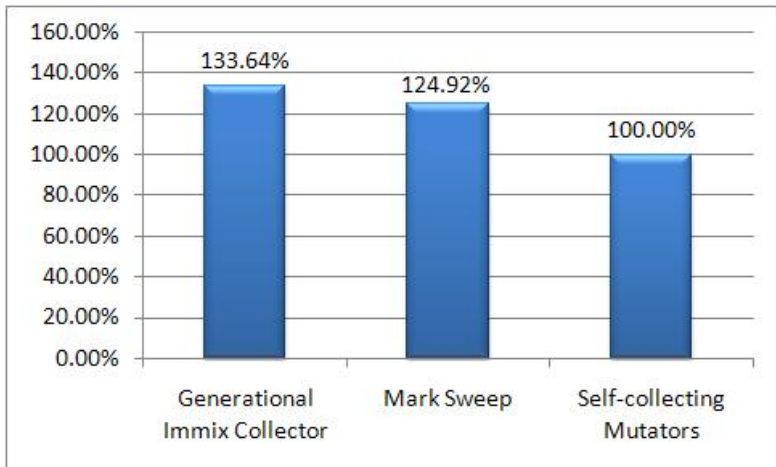
- Initialization Phase
- Main Loop
- Finalization Phase

- Ubuntu 9.10
- Jikes 3.1.0

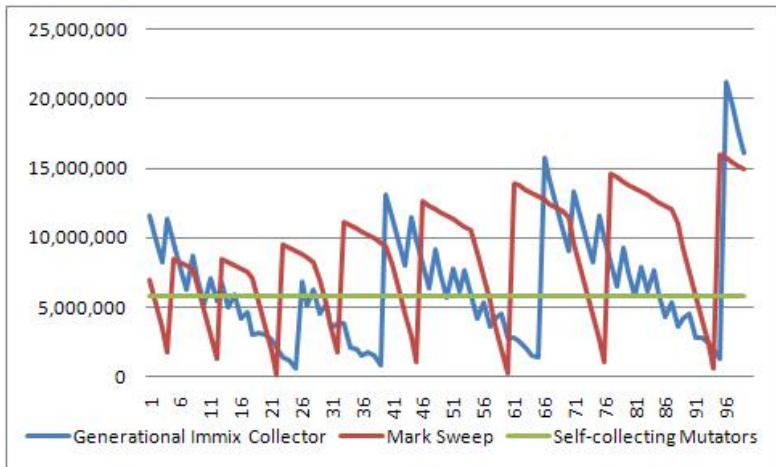
- Overall Performance
- Latency
- Memory Consumption

- Monte Carlo
  - Lines of Code: 1450
  - Chanced Line of codes: <10
- MP3 Encoder:
  - Lines of Code: 8247
  - Chanced Line of codes: 1

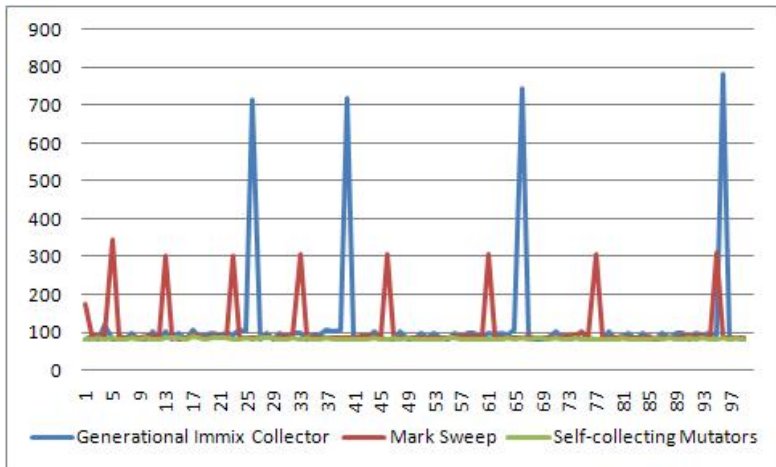
# Charts MonteCarlo Execution Time



# Charts MonteCarlo Memory Consumption

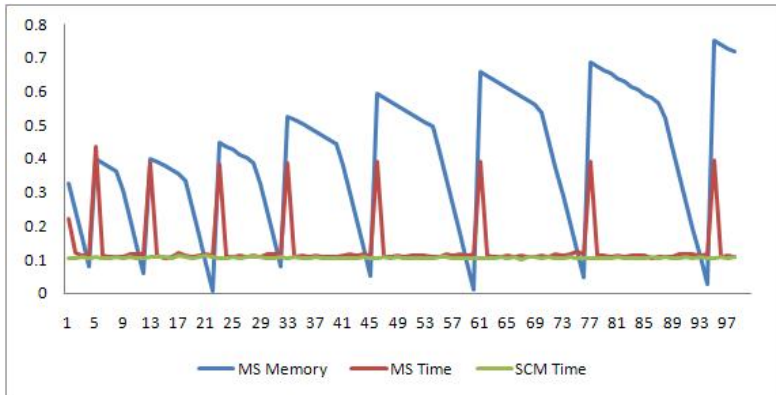


# Charts MonteCarlo Latency

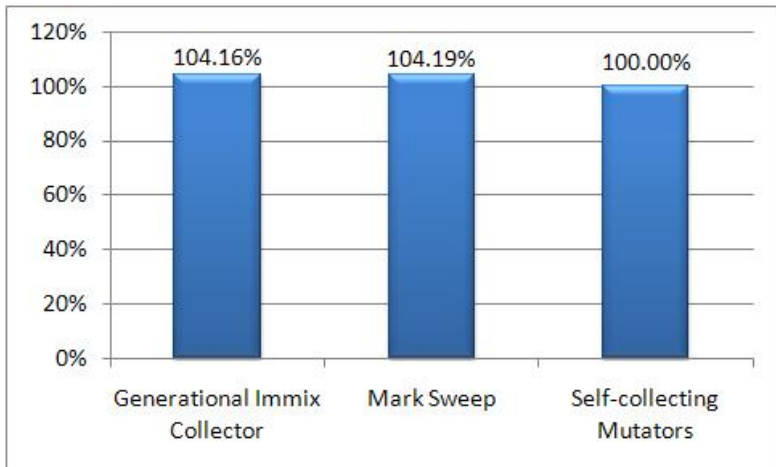




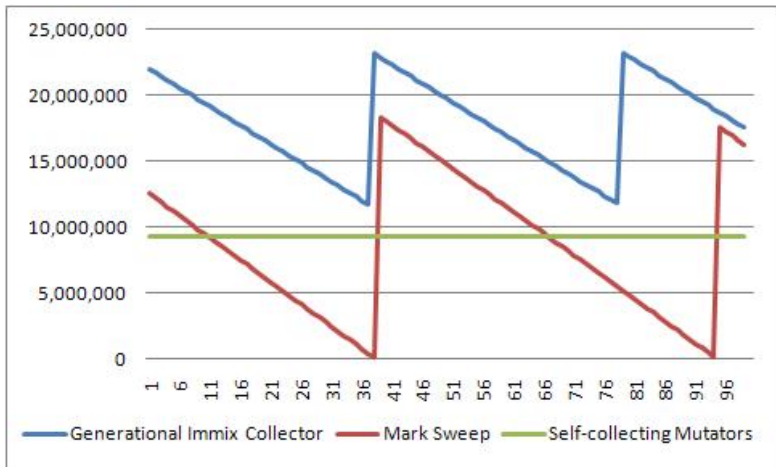
# Charts MarksSweep Memory - Latency



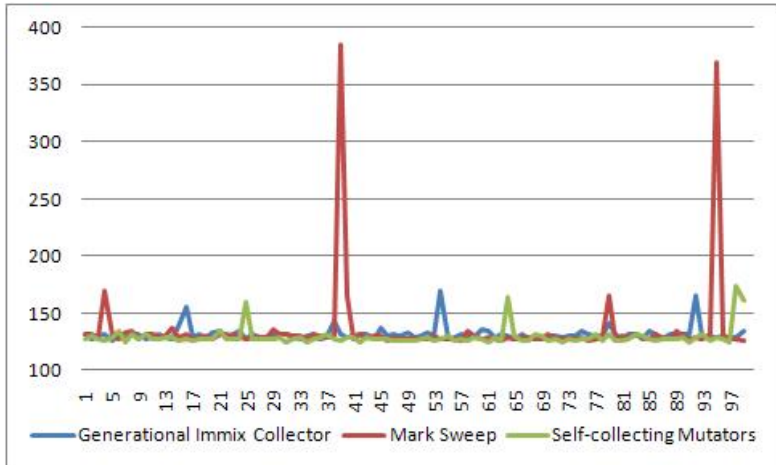
# Charts JLayer Execution Time



# Charts JLayer Memory Consumption



# Charts JLayer Latency



Danke für eure Aufmerksamkeit