HEAP EVOLUTION ANALYSIS USING TREE VISUALIZATIONS

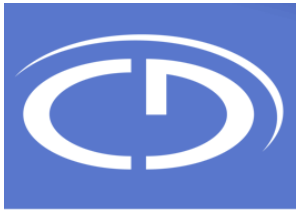
SSP 2020

Markus Weninger, Lukas Makor, Hanspeter Mössenböck

Johannes Kepler University Linz, Austria

Institute for System Software, Christian Doppler Laboratory MEVSS

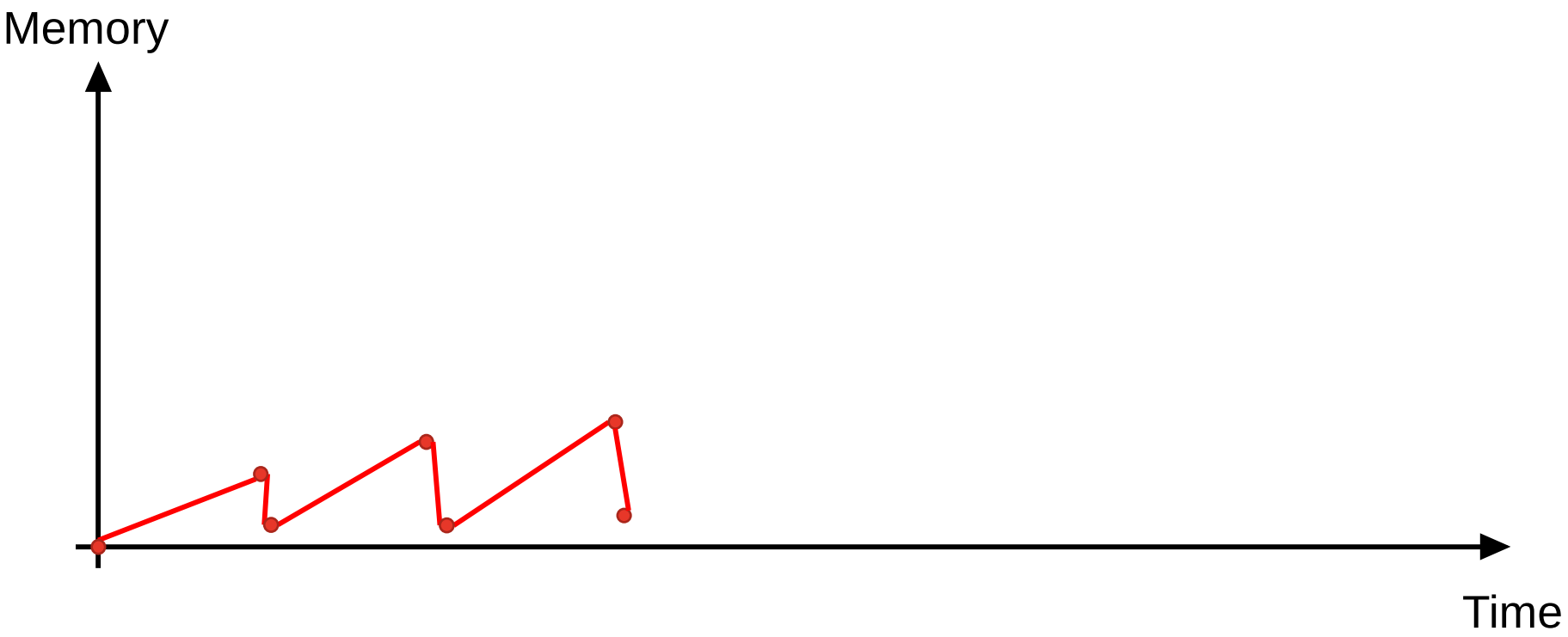
JOHANNES KEPLER
UNIVERSITY LINZ



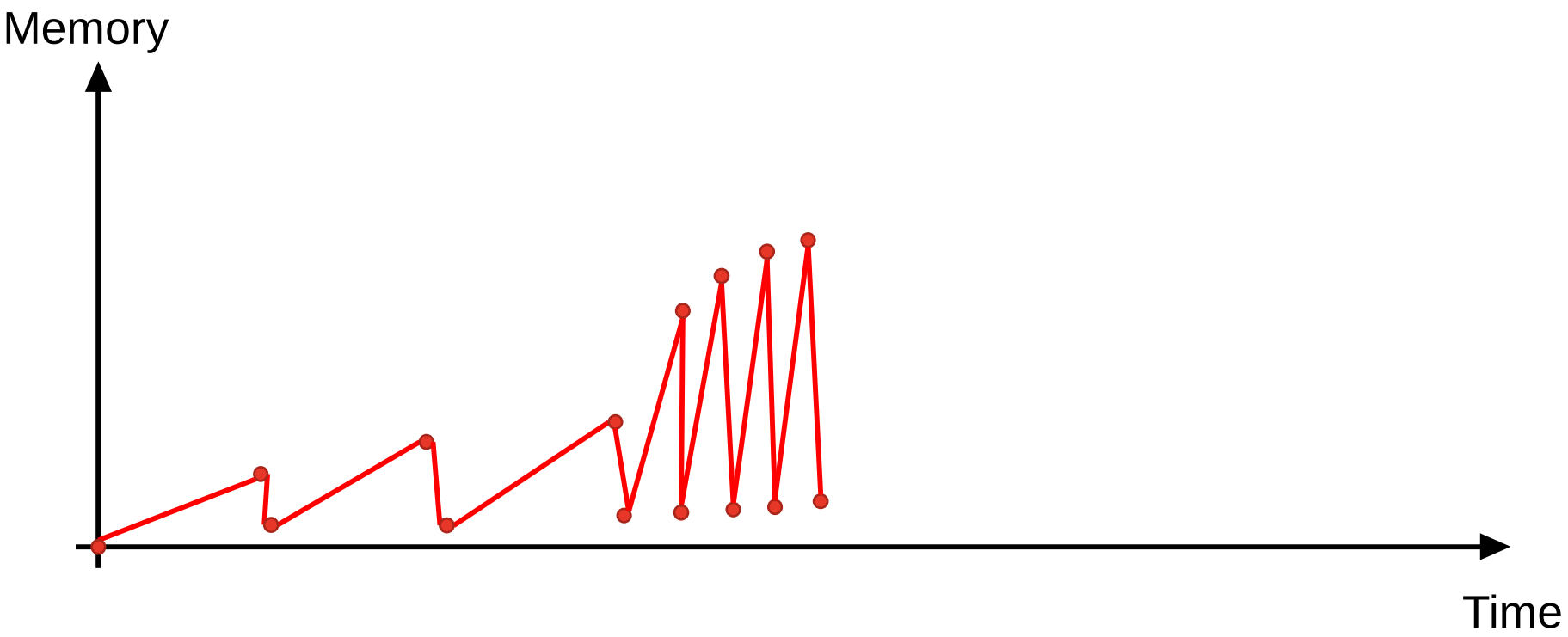
dynatrace

MOTIVATION: MEMORY ANOMALIES

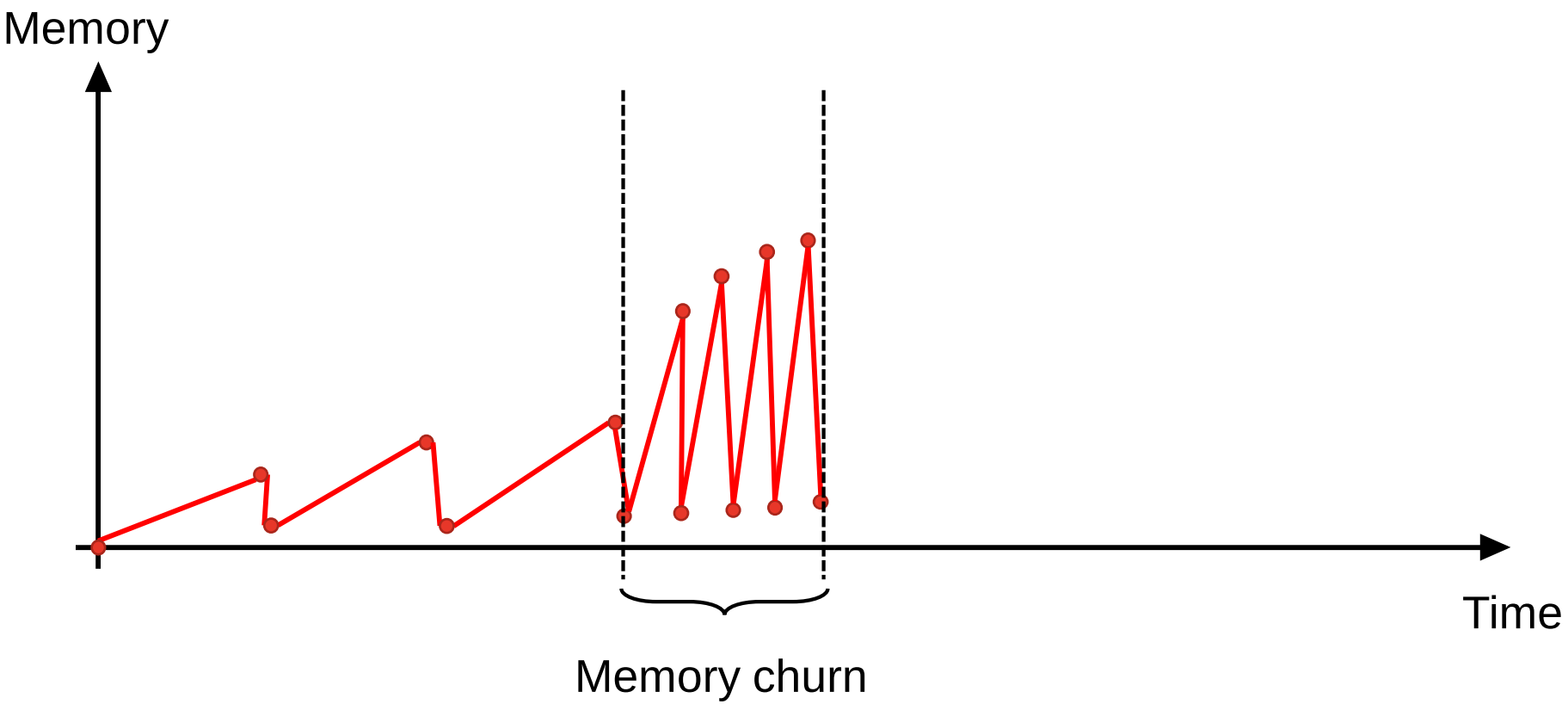
MOTIVATION: MEMORY ANOMALIES



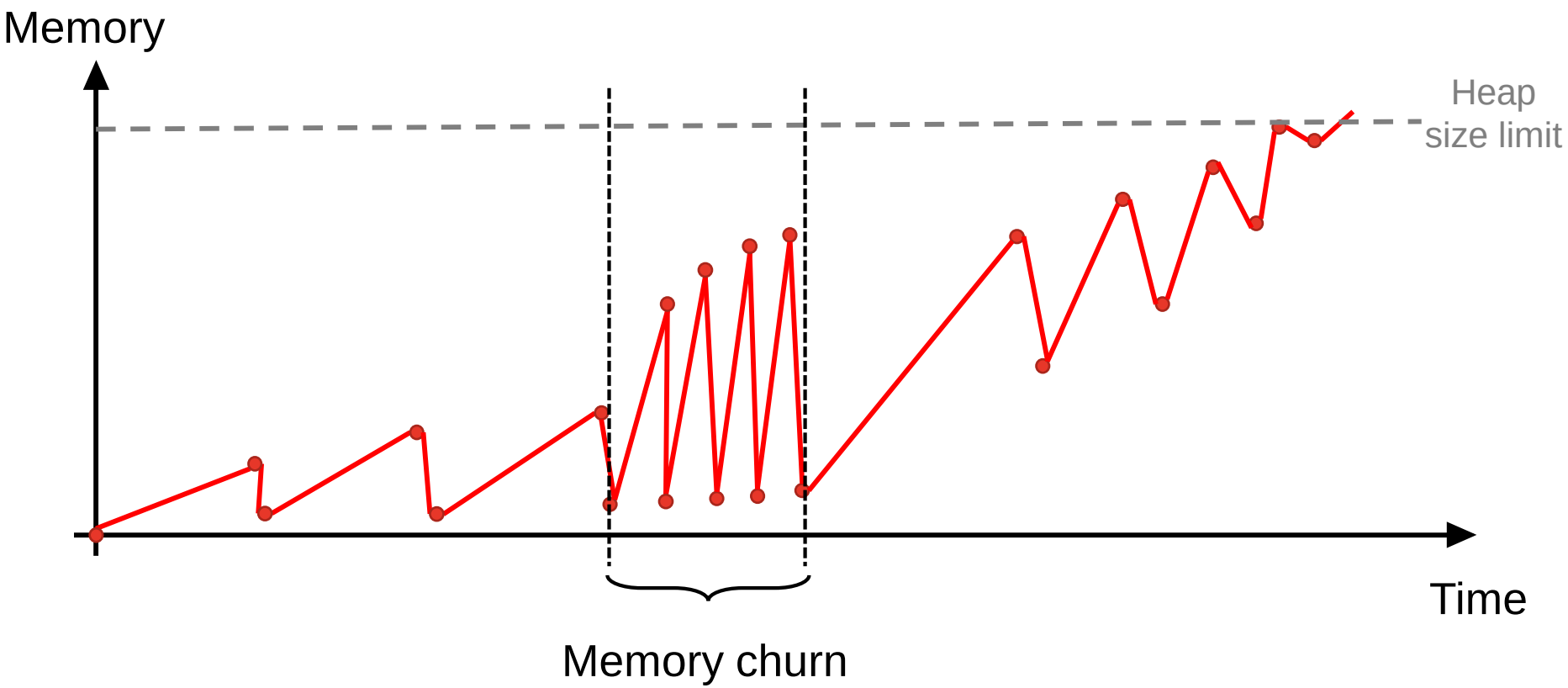
MOTIVATION: MEMORY ANOMALIES



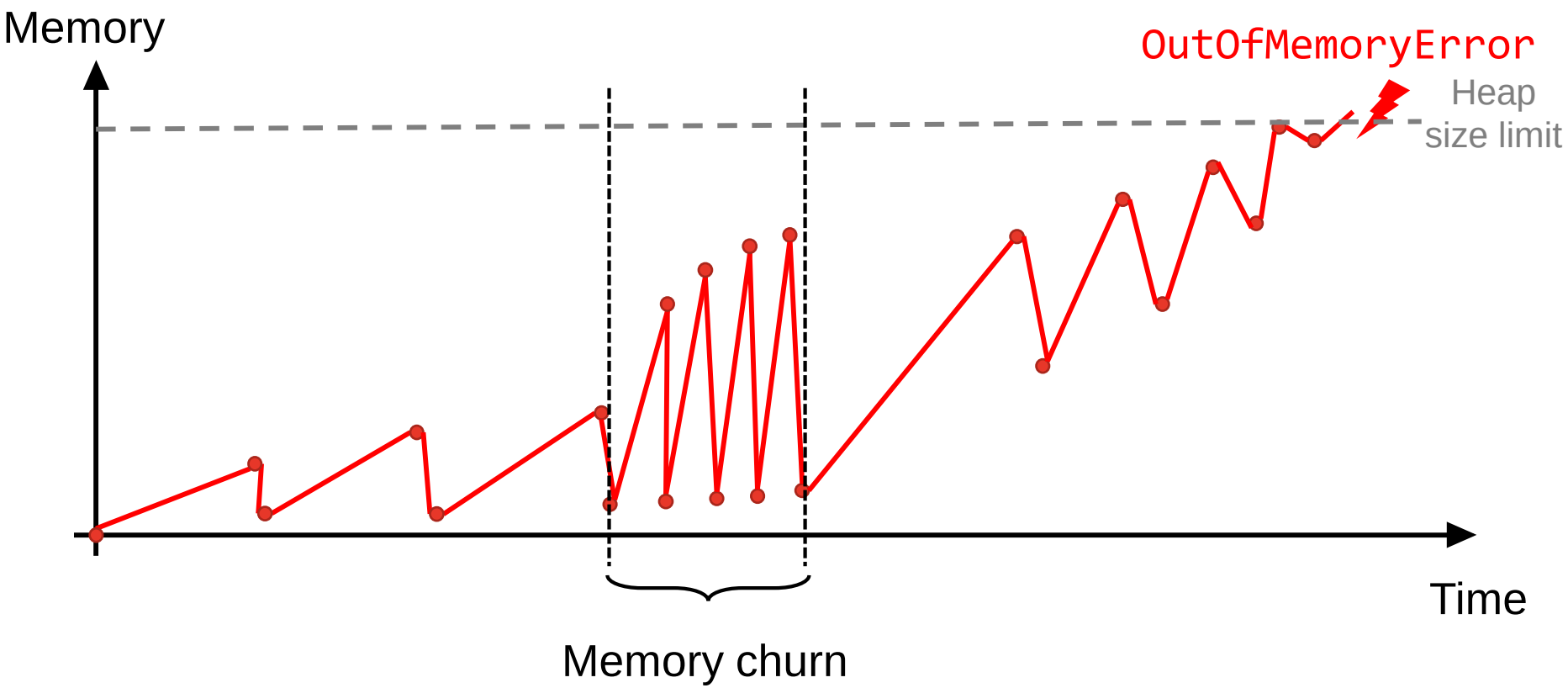
MOTIVATION: MEMORY ANOMALIES



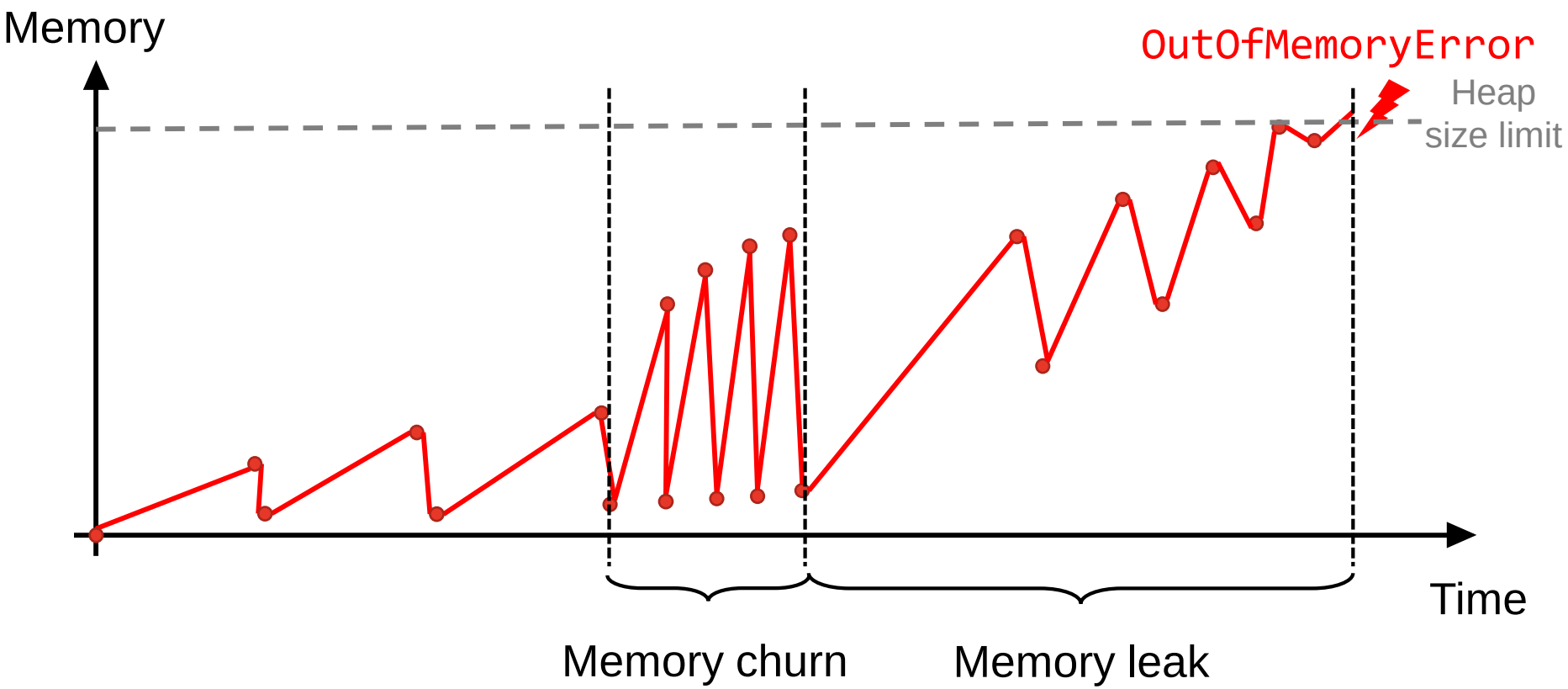
MOTIVATION: MEMORY ANOMALIES



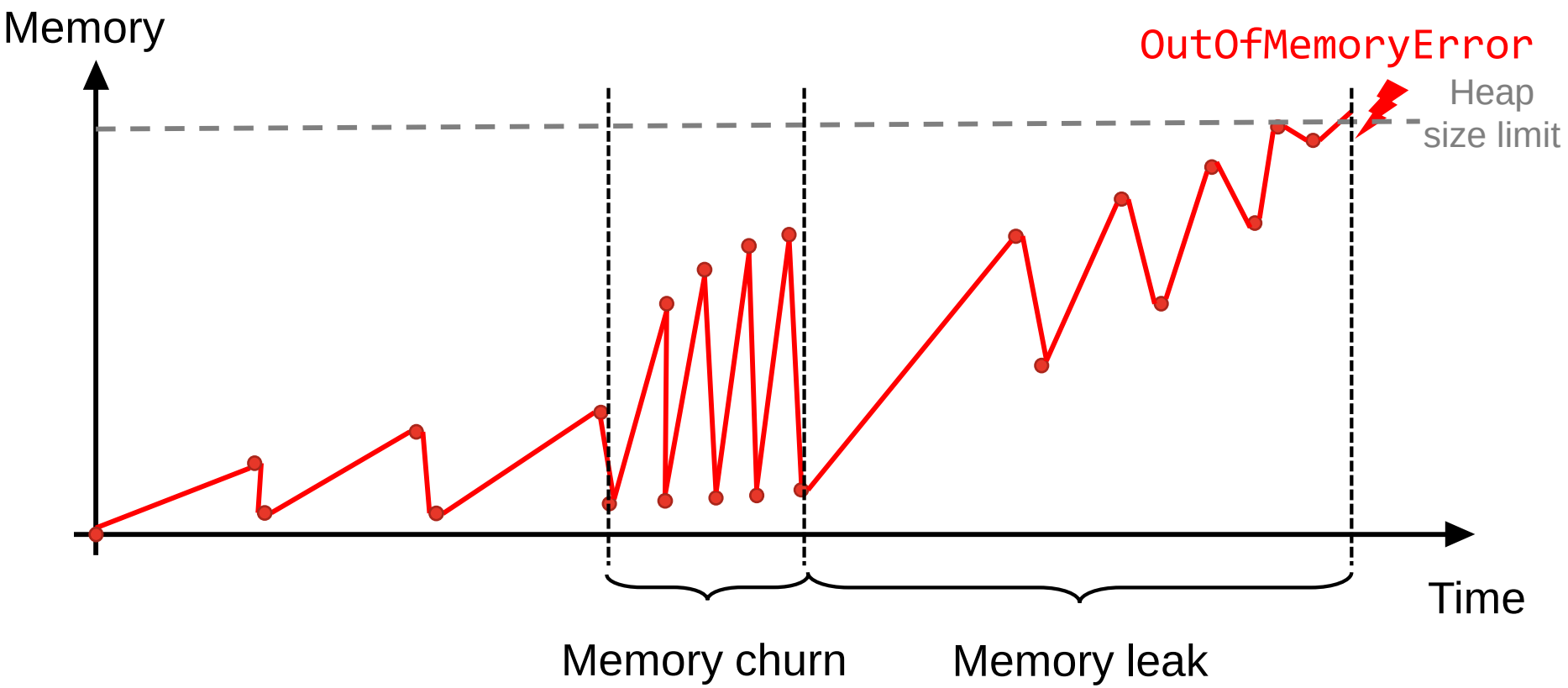
MOTIVATION: MEMORY ANOMALIES



MOTIVATION: MEMORY ANOMALIES

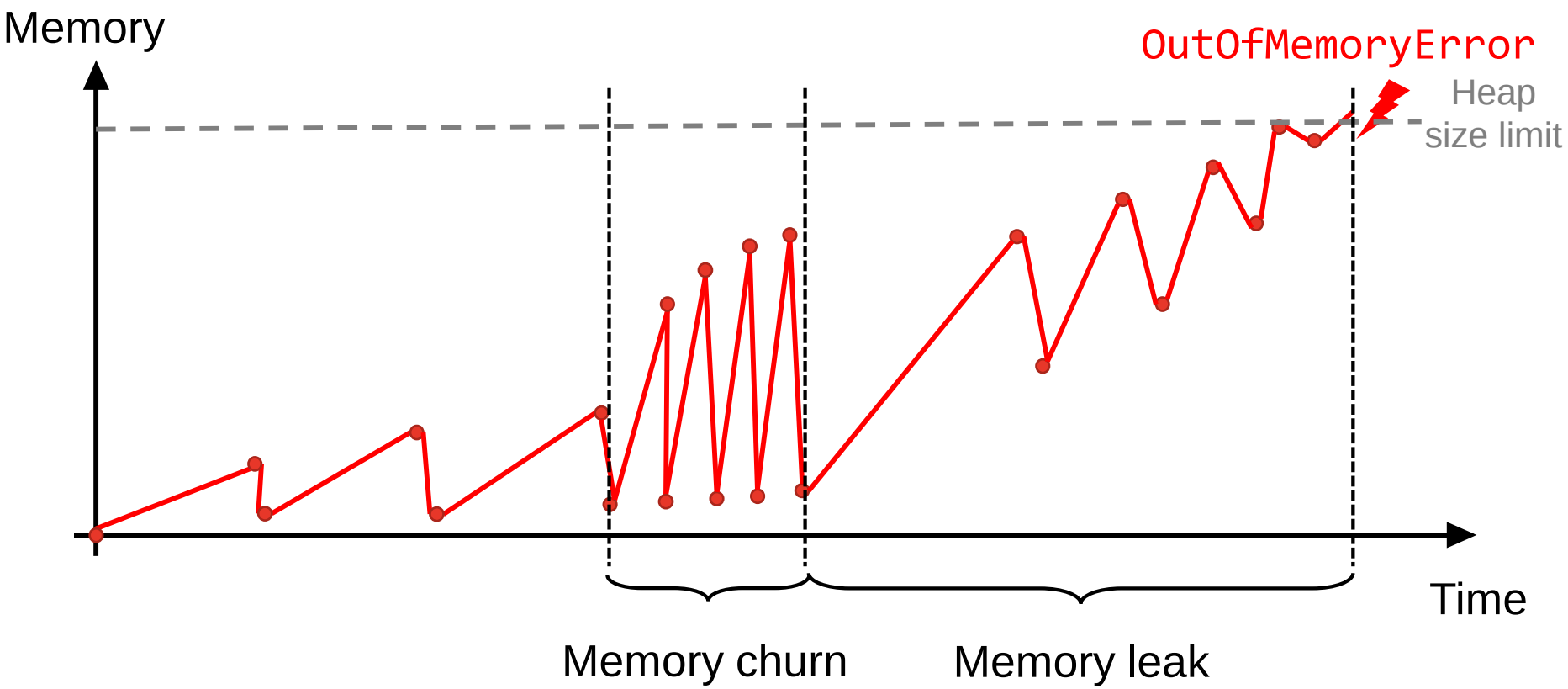


MOTIVATION: MEMORY ANOMALIES



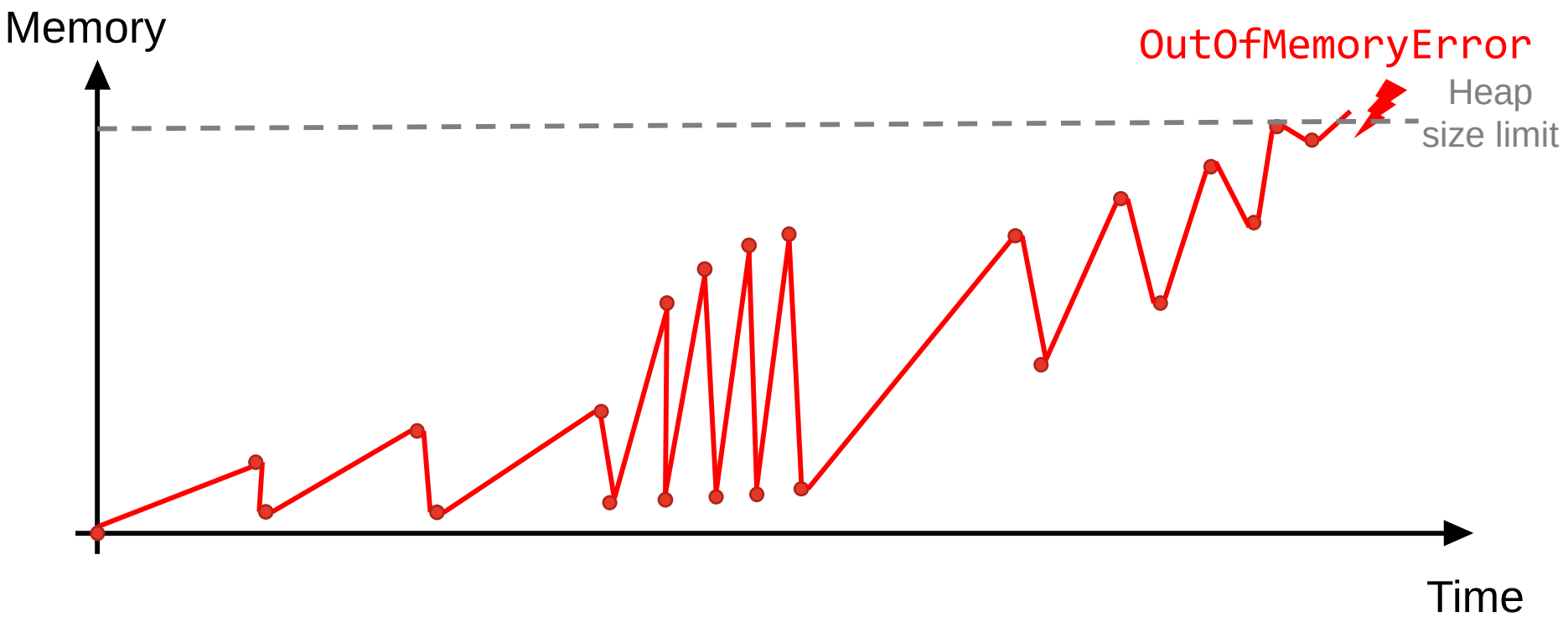
Investigate!

MOTIVATION: MEMORY ANOMALIES

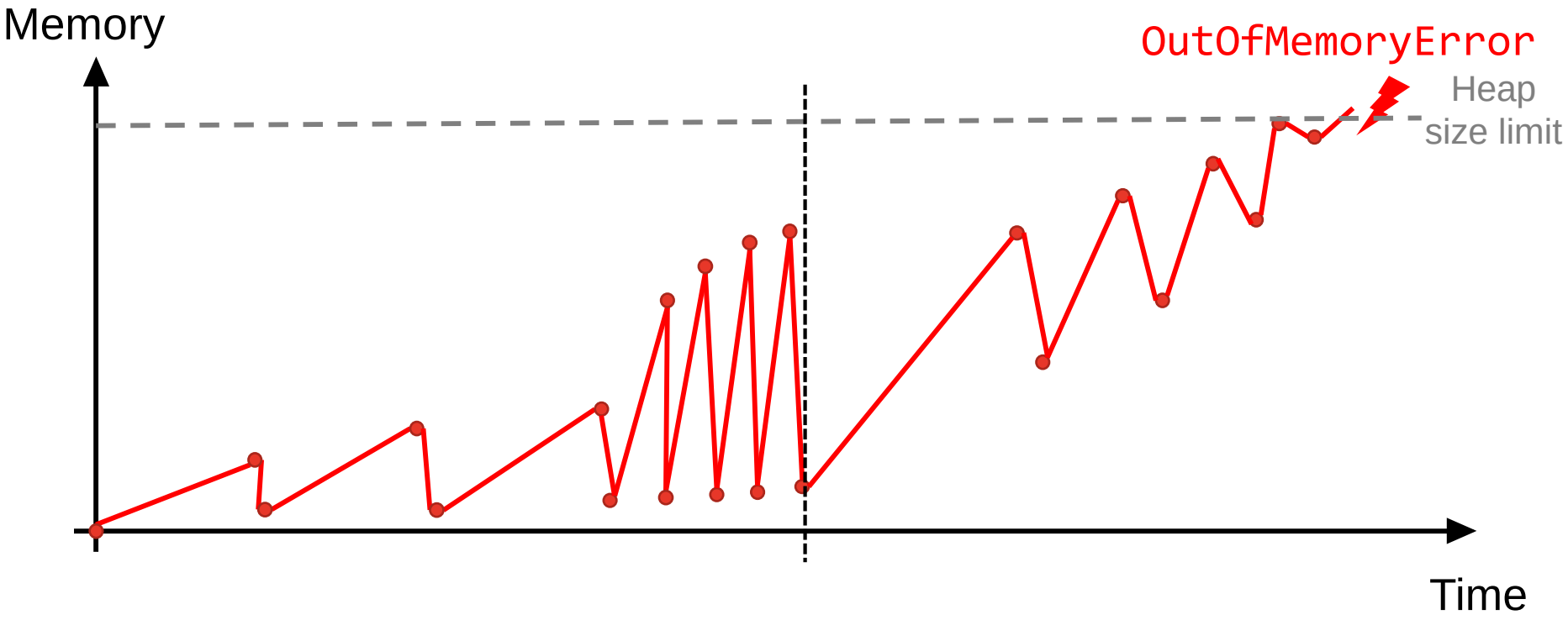


How?

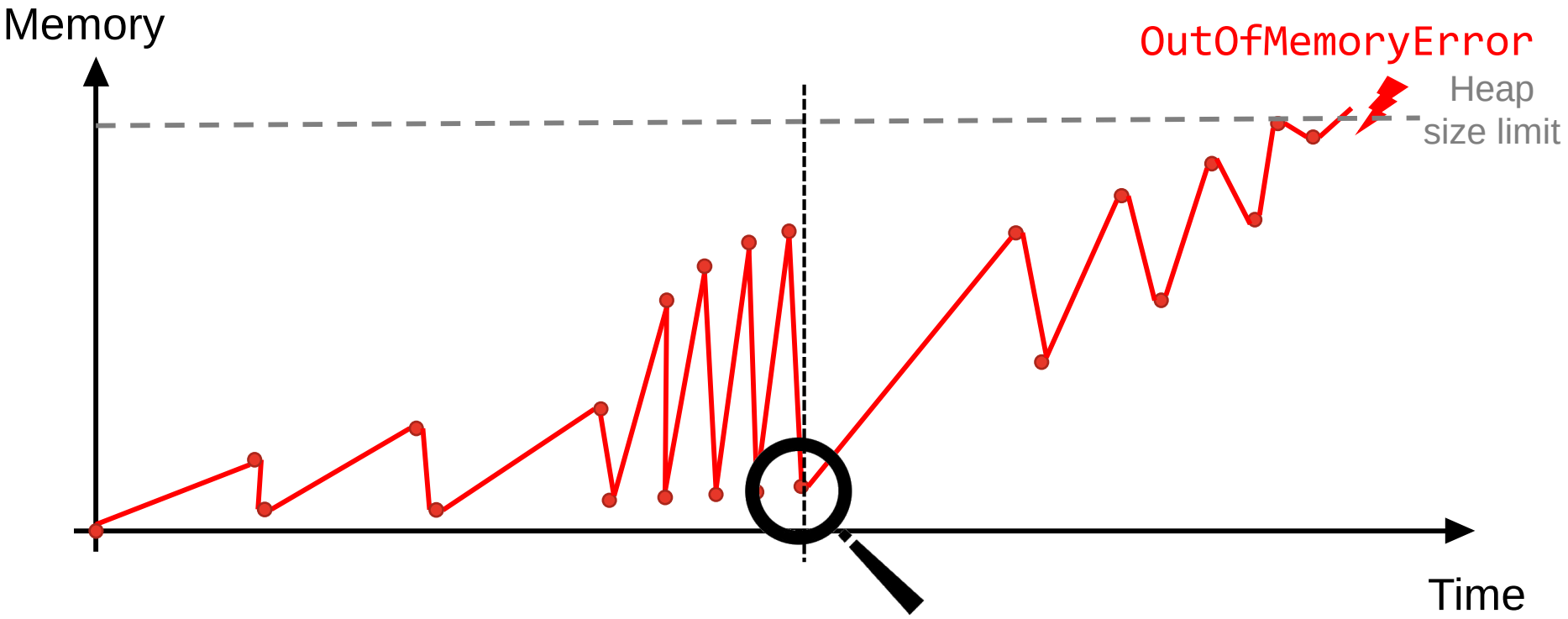
MOTIVATION: MEMORY ANOMALIES



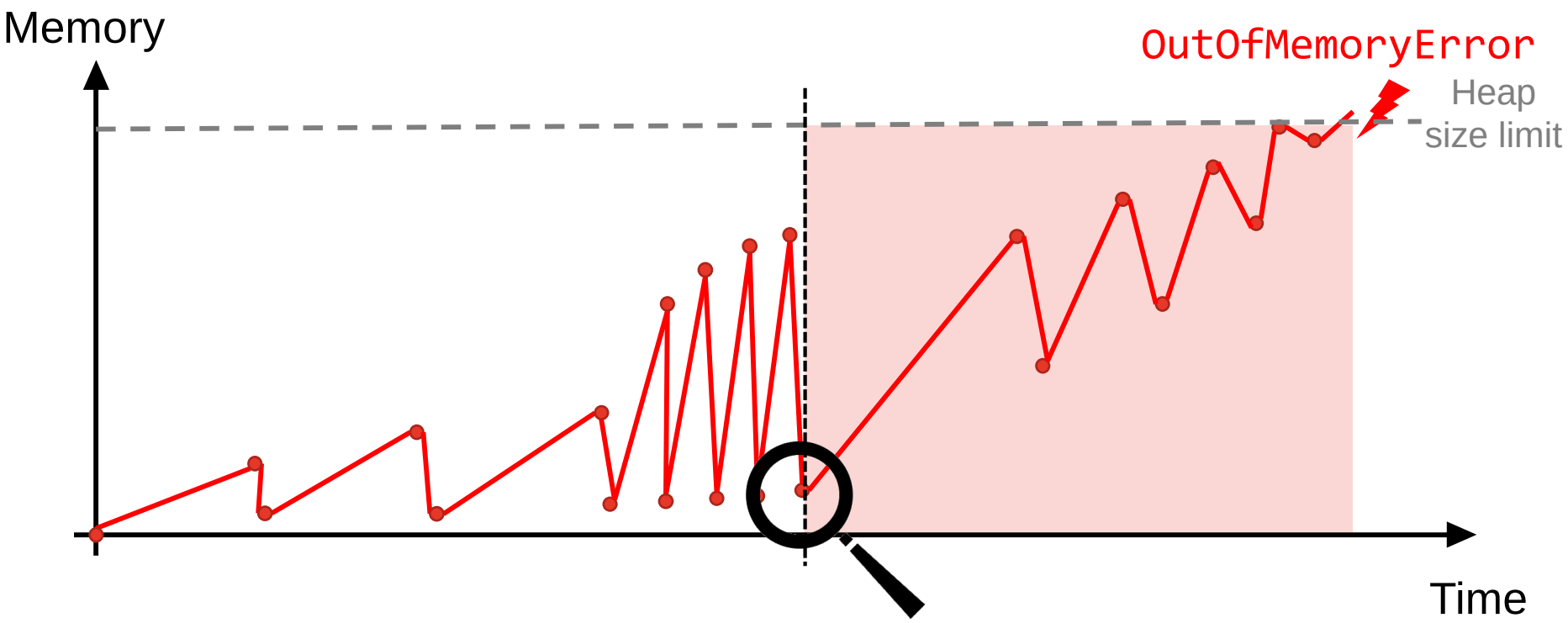
MOTIVATION: MEMORY ANOMALIES



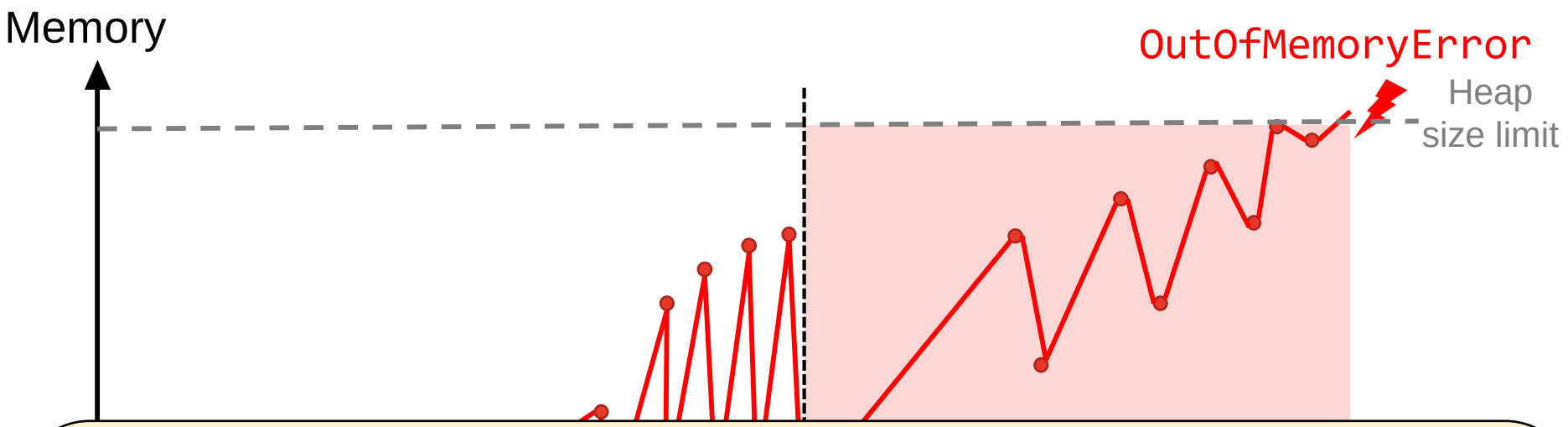
MOTIVATION: MEMORY ANOMALIES



MOTIVATION: MEMORY ANOMALIES

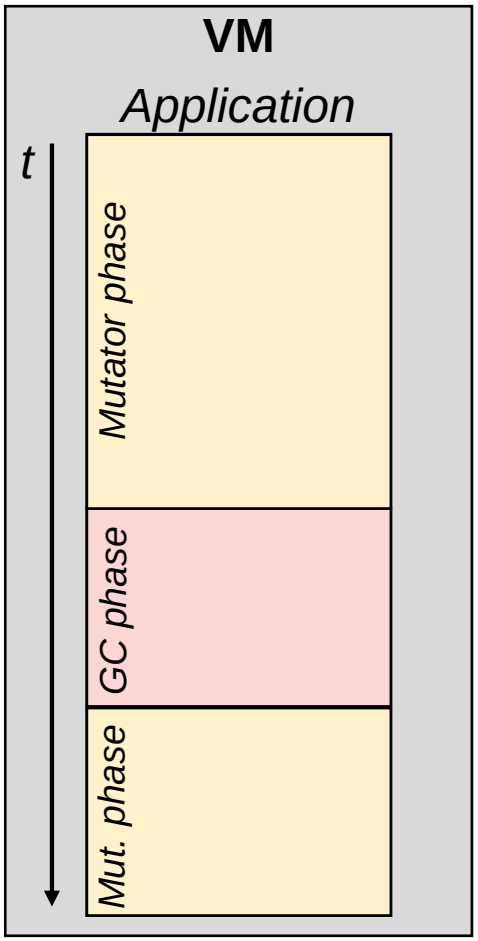


MOTIVATION: MEMORY ANOMALIES



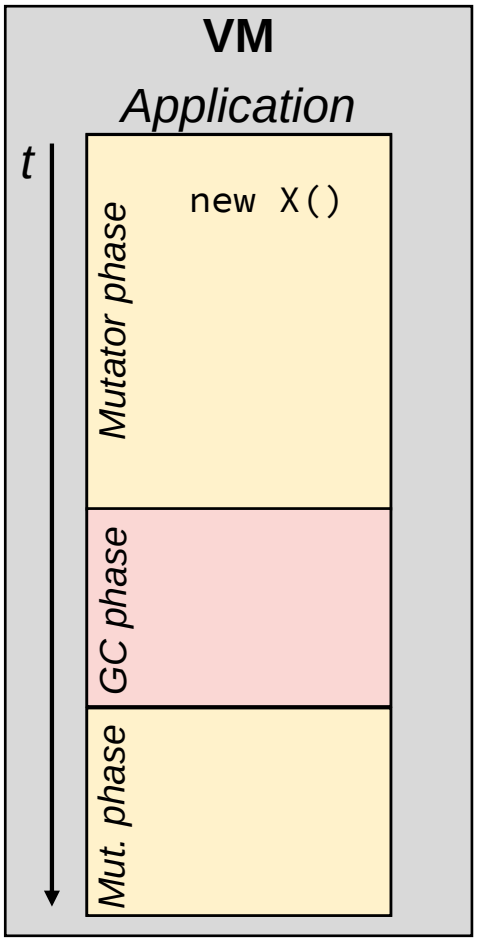
Idea: Help developers in ***understanding*** memory ***growth*** over time through the means of ***visualization***

DATA: MEMORY TRACES



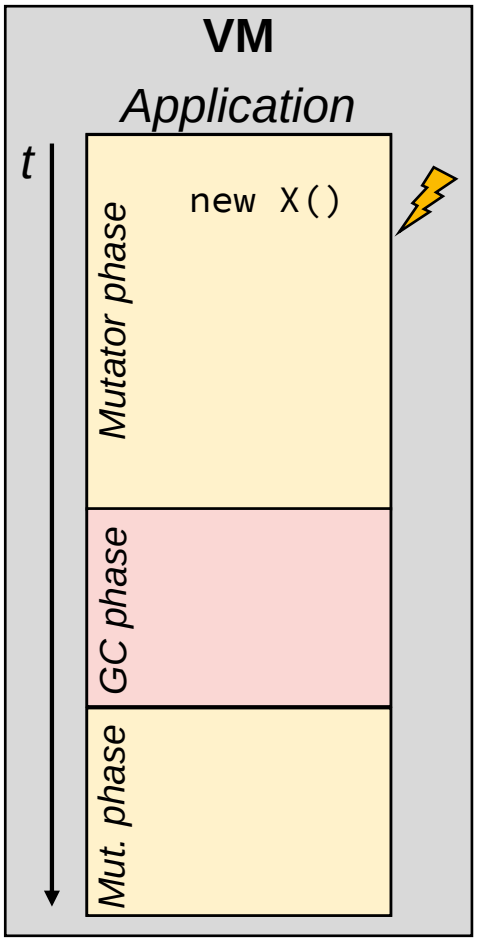
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



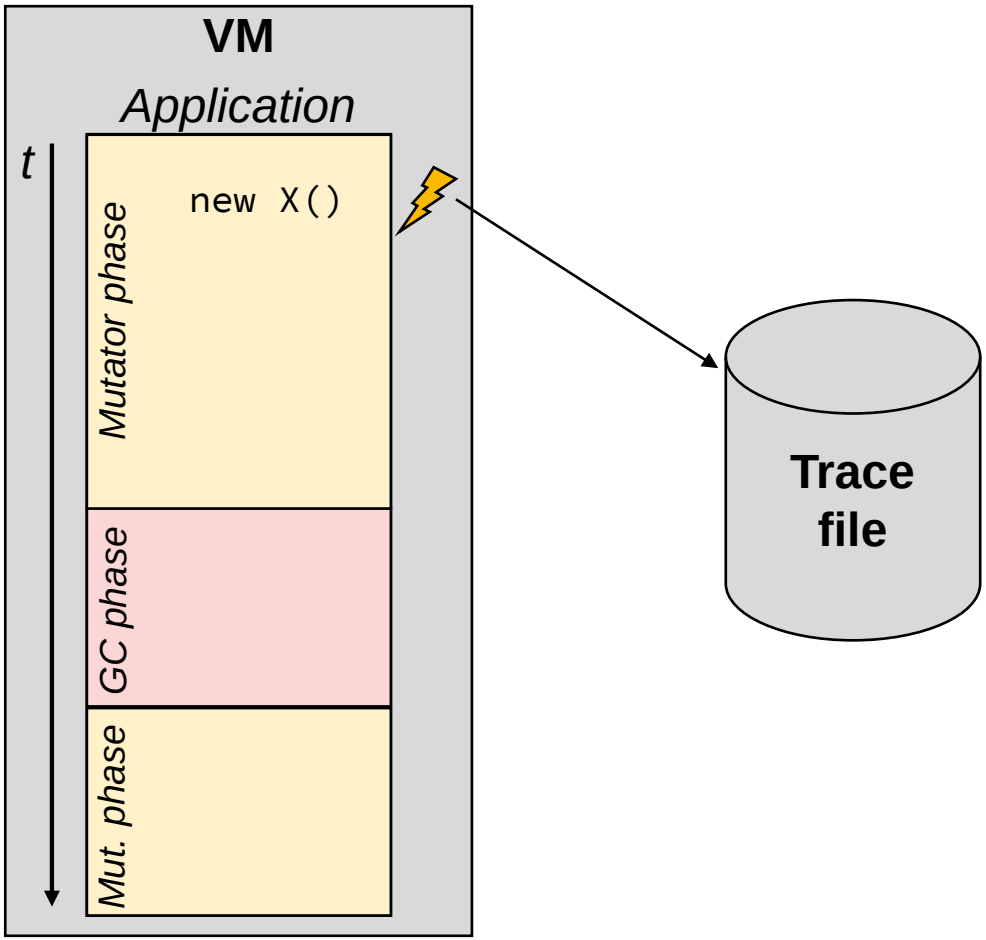
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



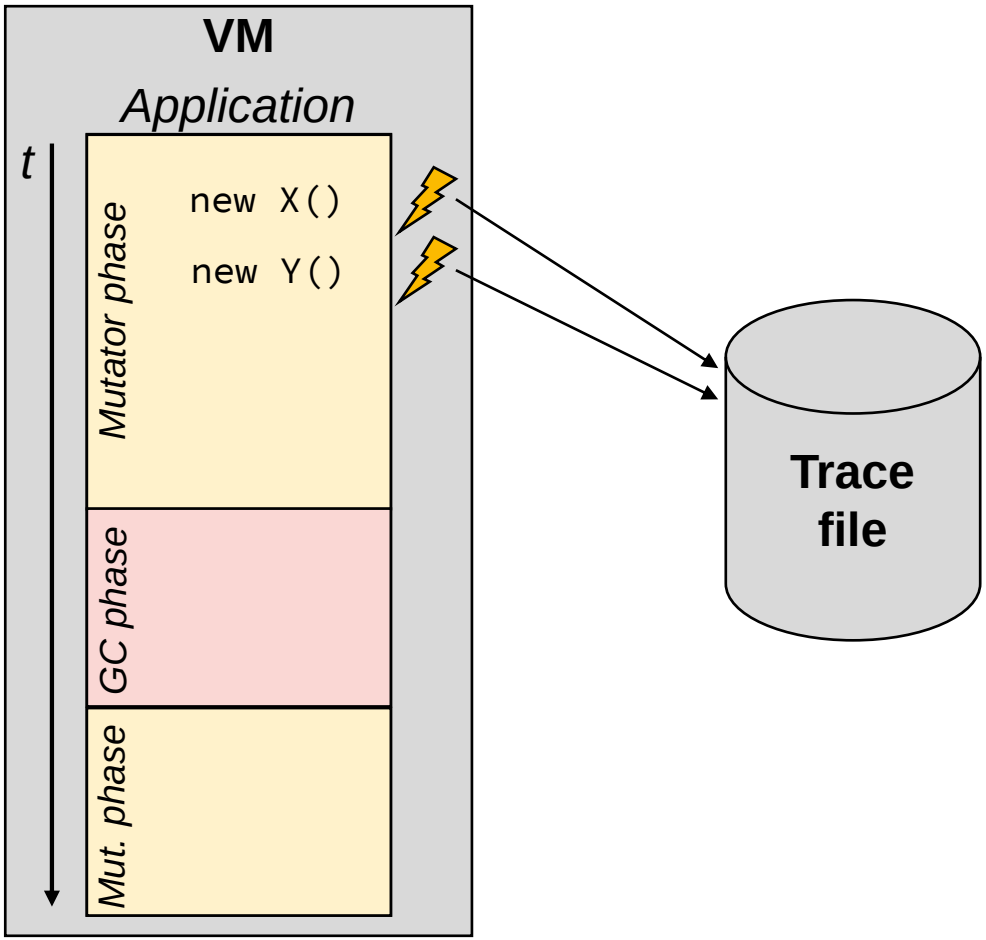
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



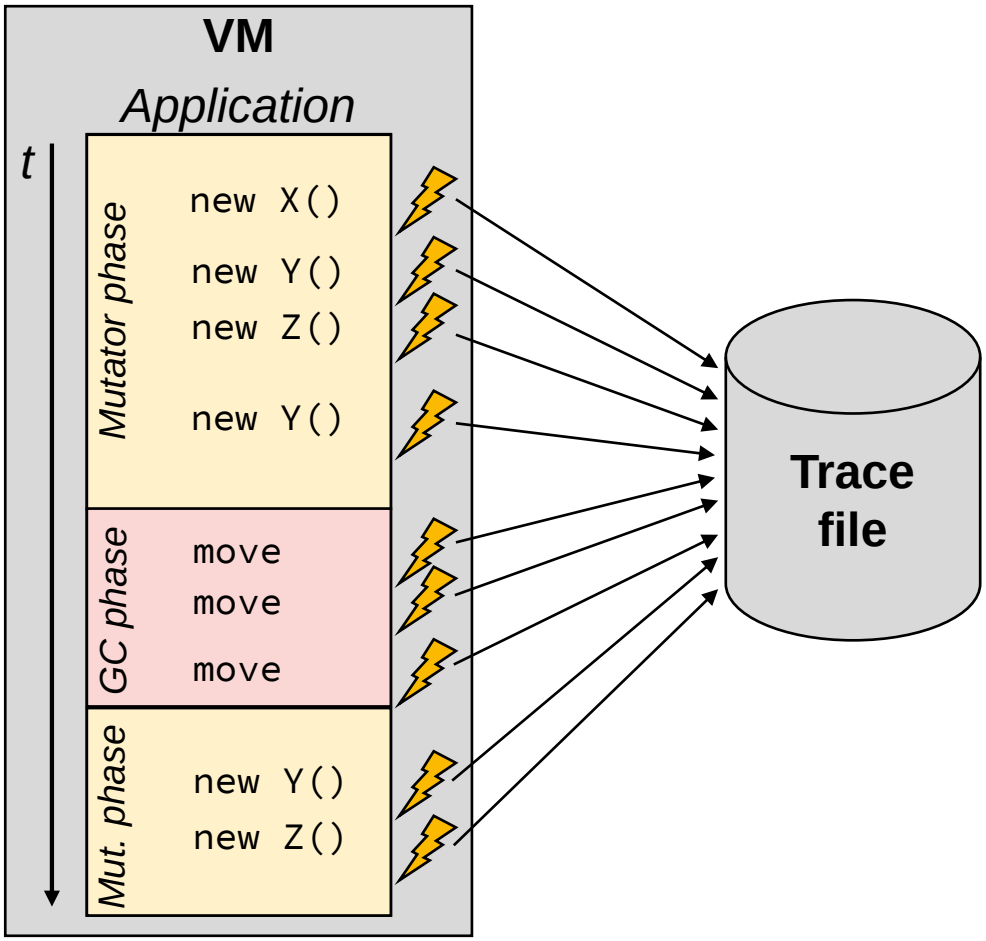
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



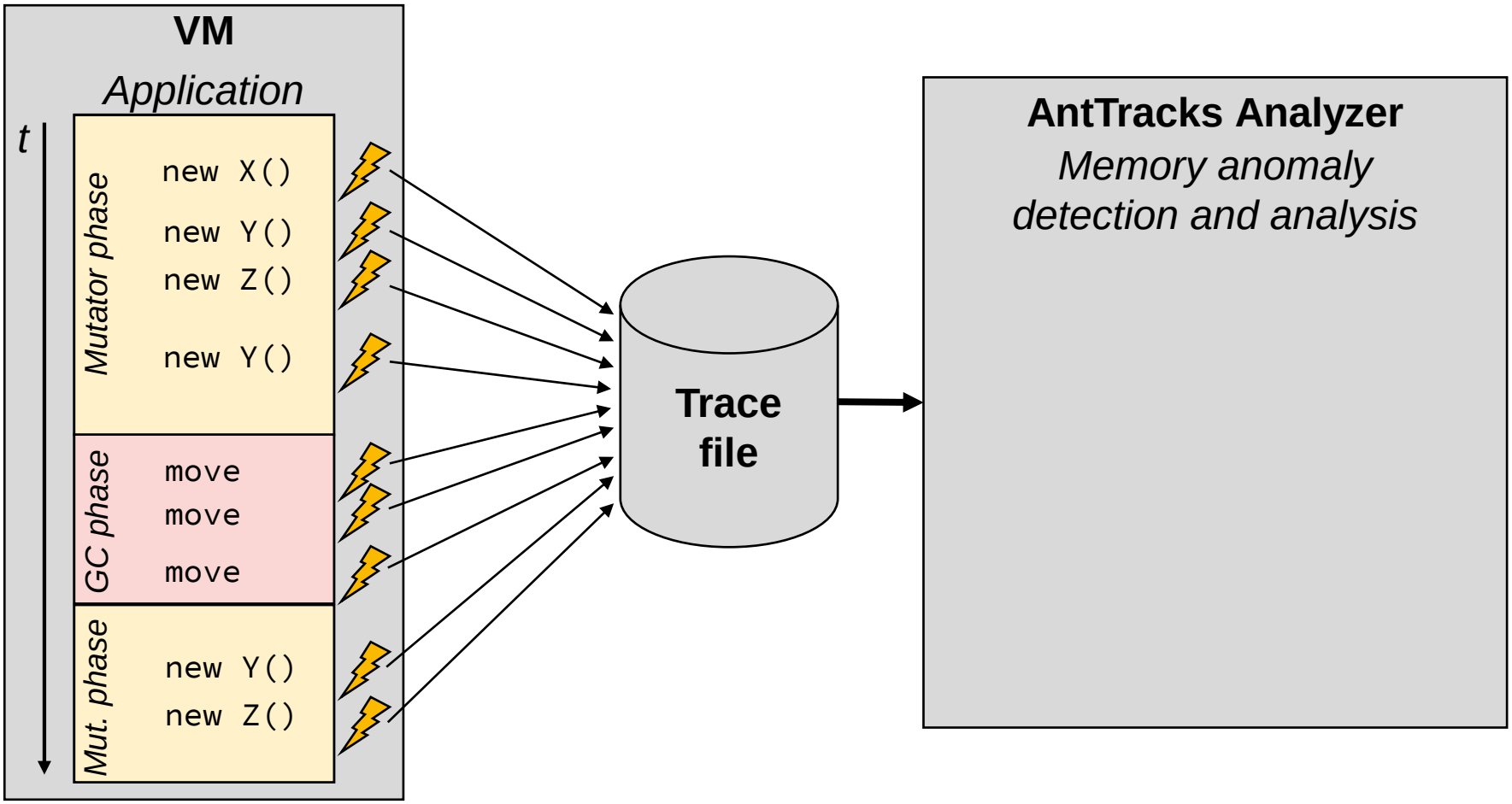
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



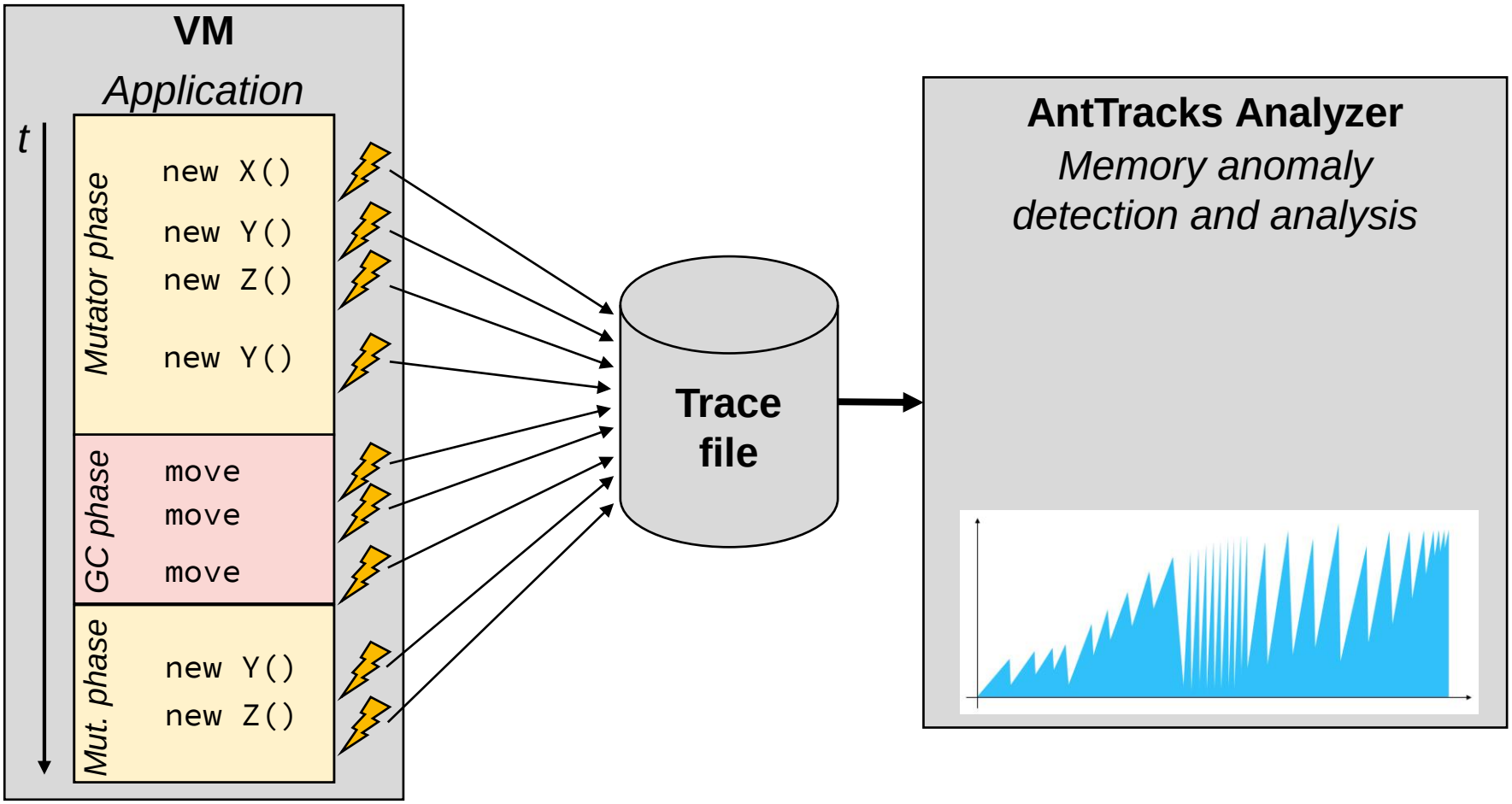
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



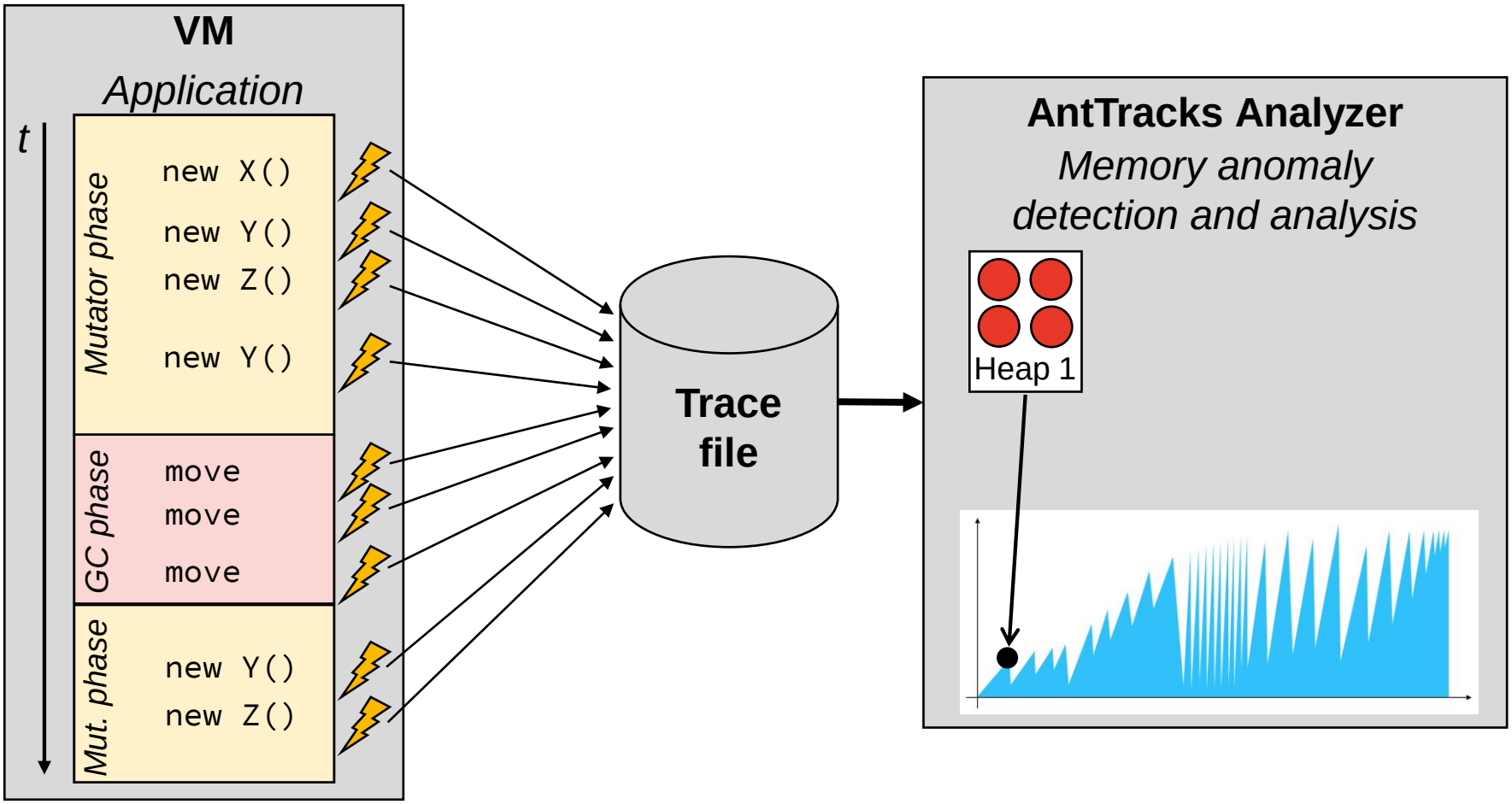
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



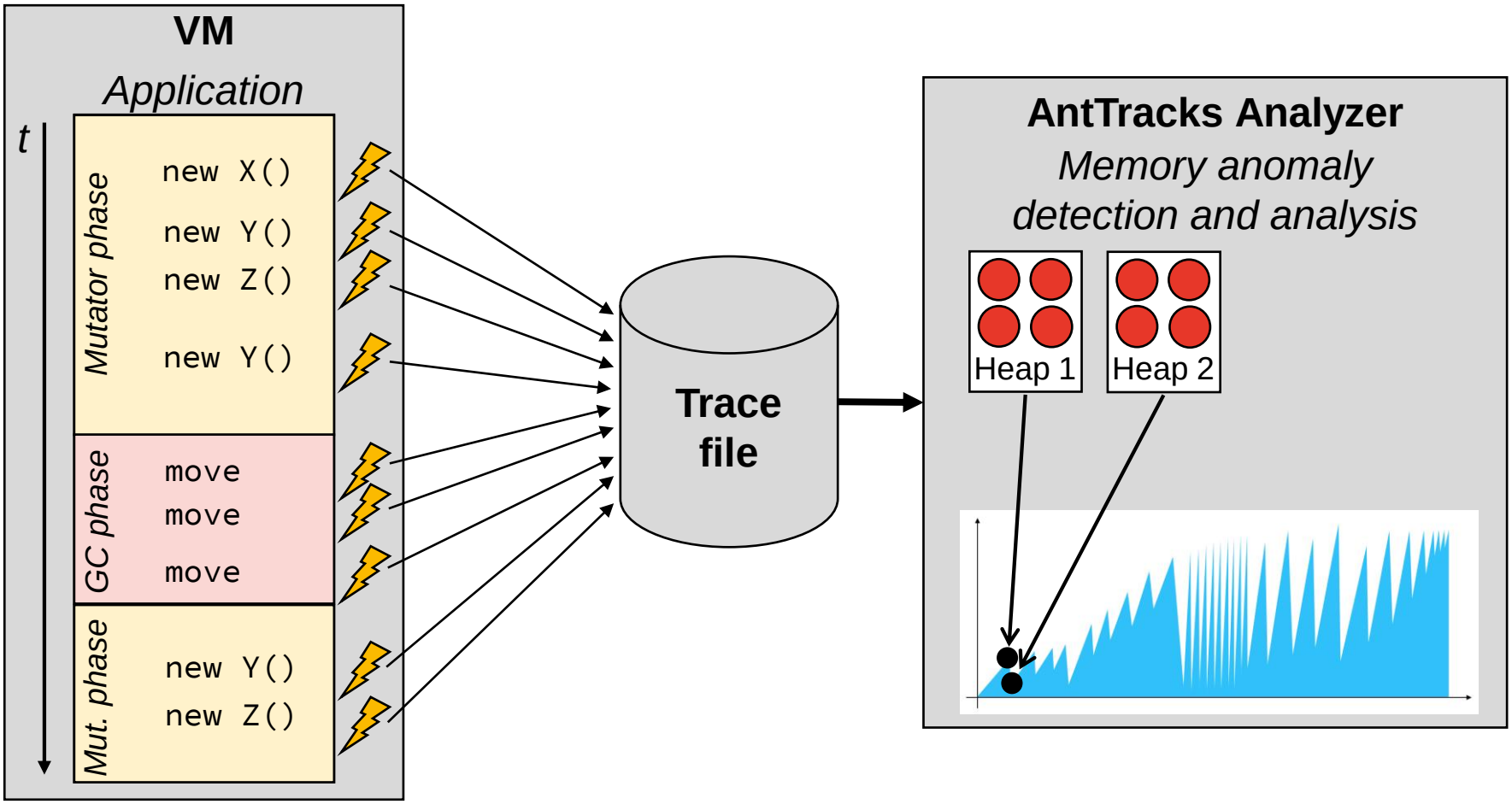
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



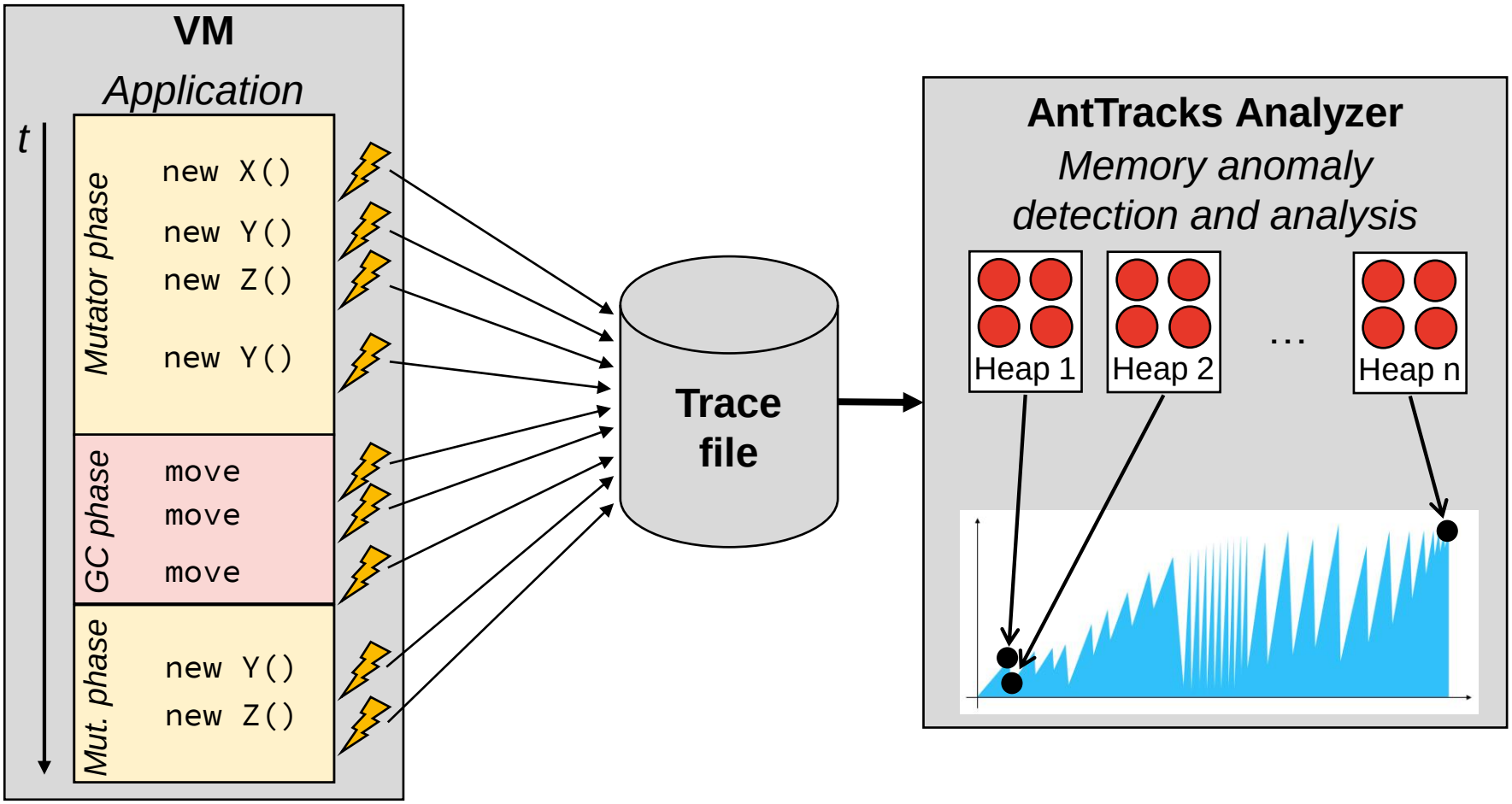
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



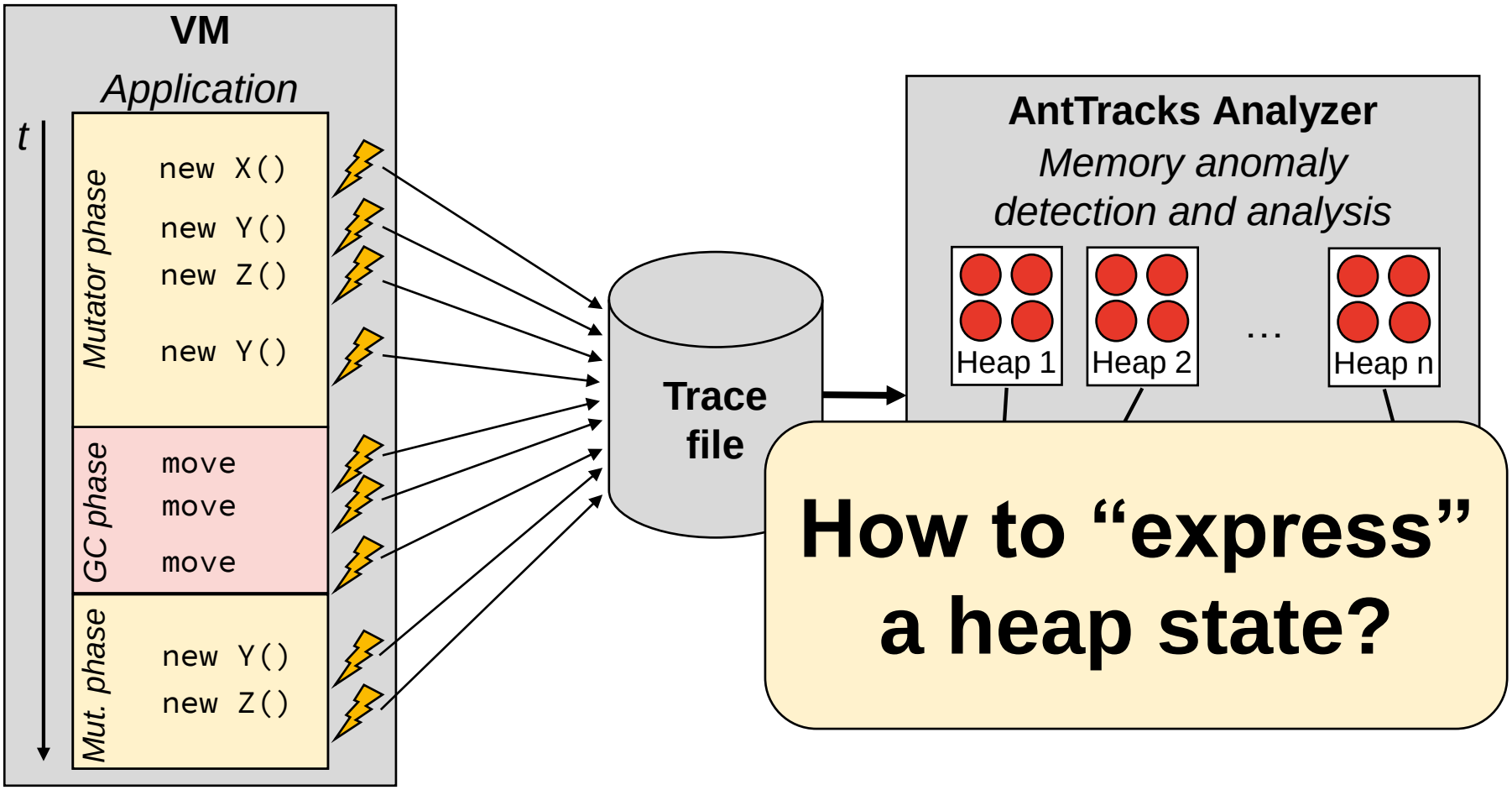
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

DATA: MEMORY TRACES



Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

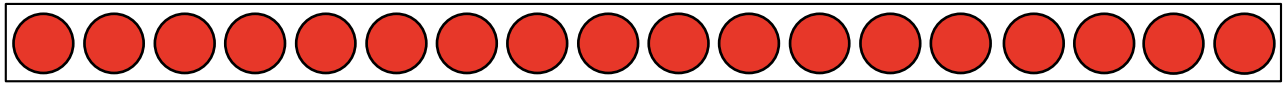
DATA: MEMORY TRACES



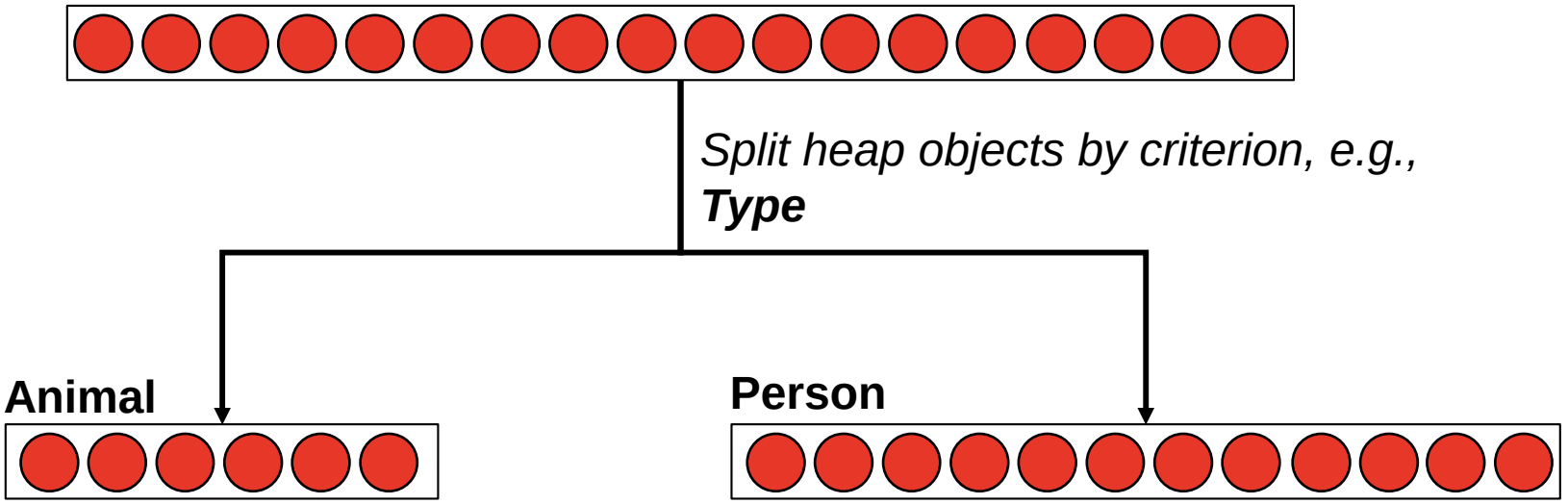
Lengauer, Bitto, Mössenböck: Accurate and Efficient Object Tracing for Java Applications, ICPE 2015
Lengauer et al.: Efficient Memory Traces with Full Pointer Information, PPPJ 2016

OBJECT CLASSIFICATION

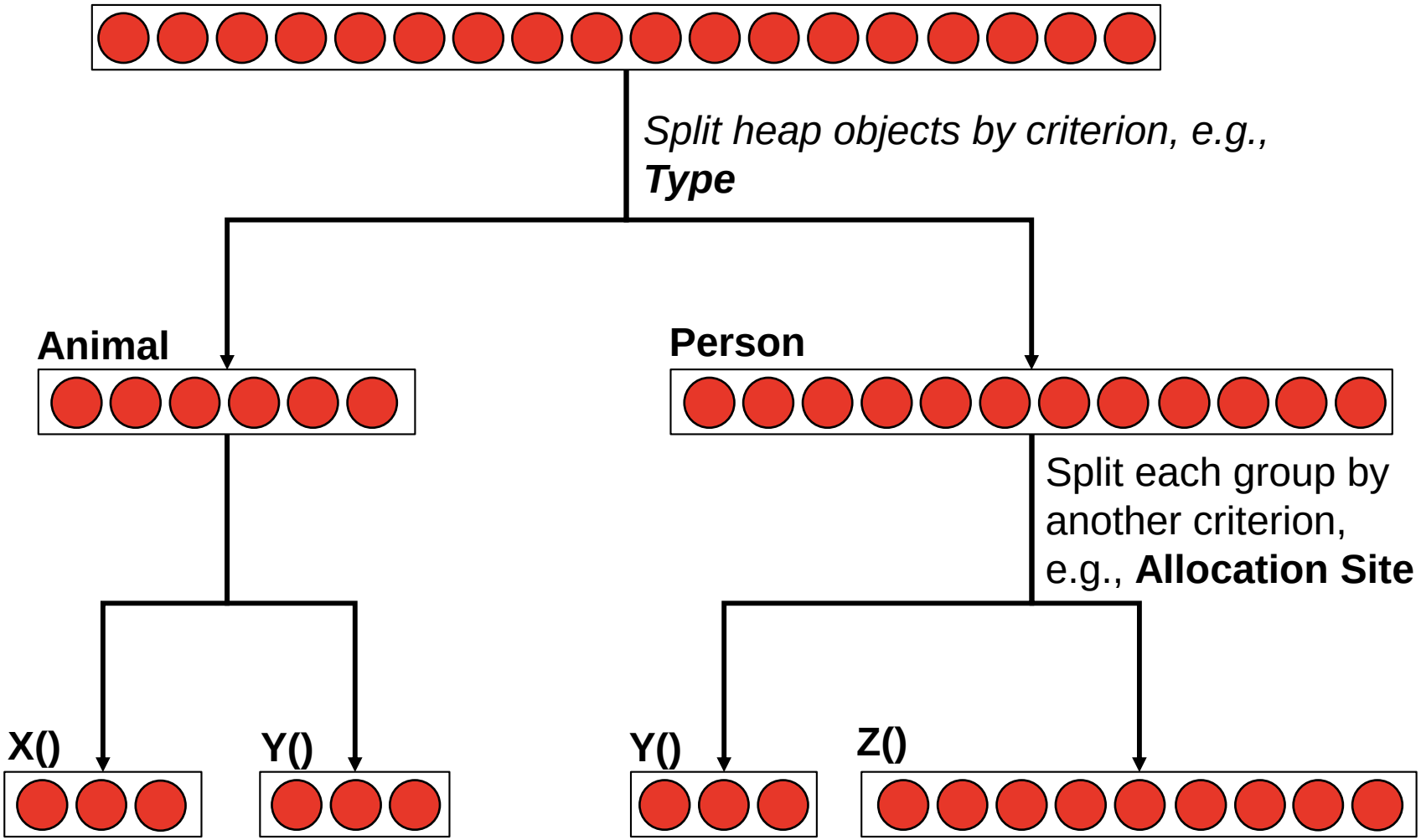
OBJECT CLASSIFICATION



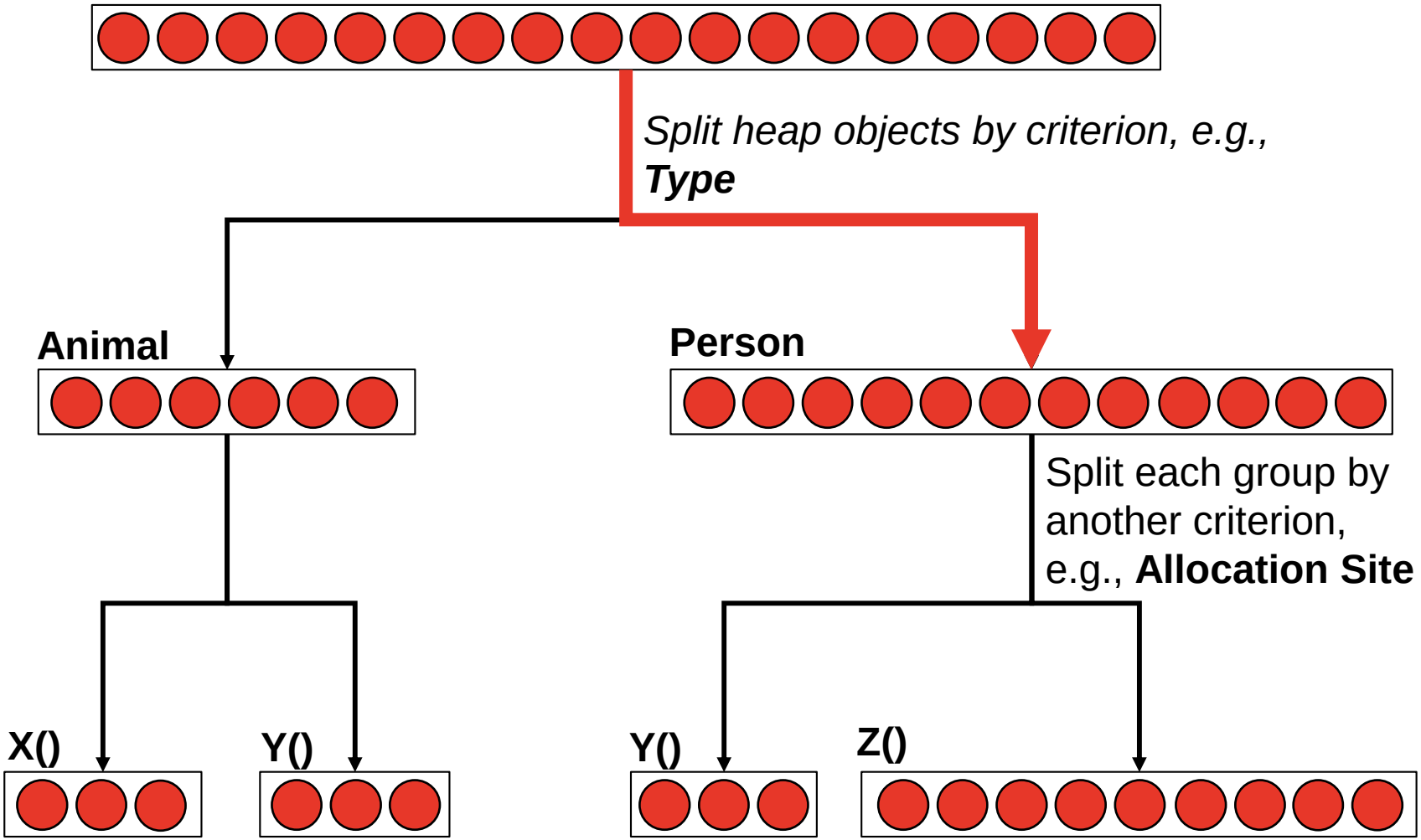
OBJECT CLASSIFICATION



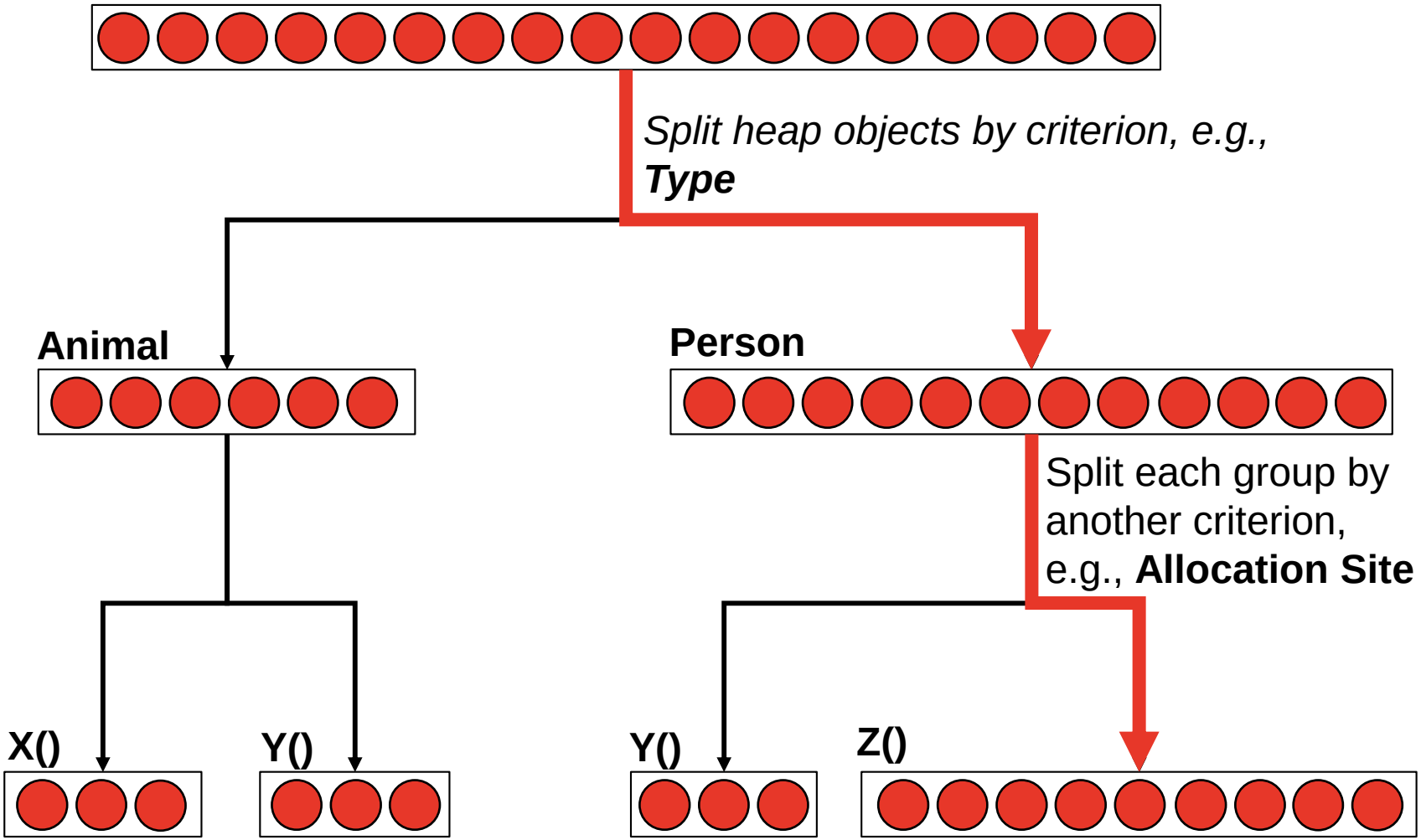
OBJECT CLASSIFICATION



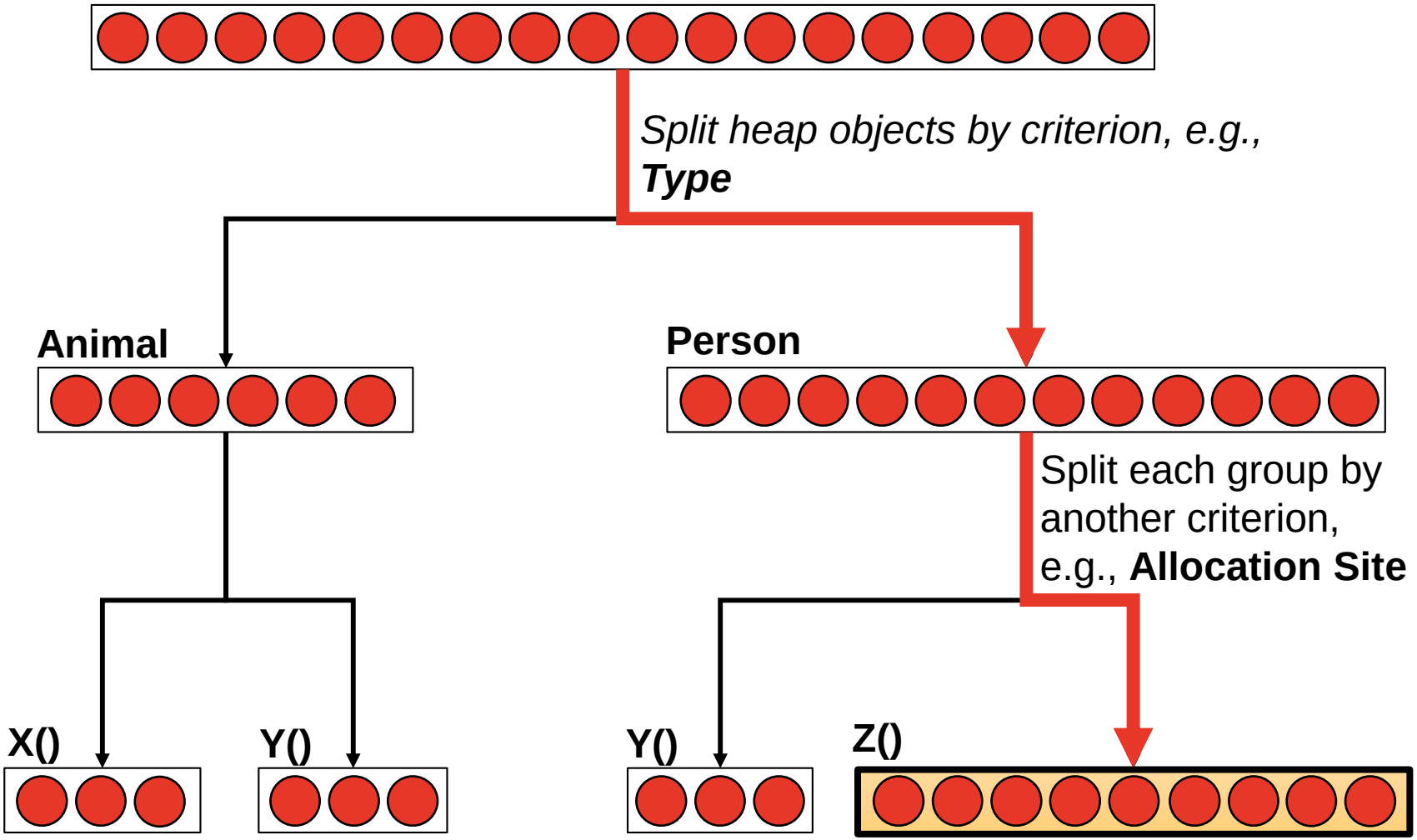
OBJECT CLASSIFICATION



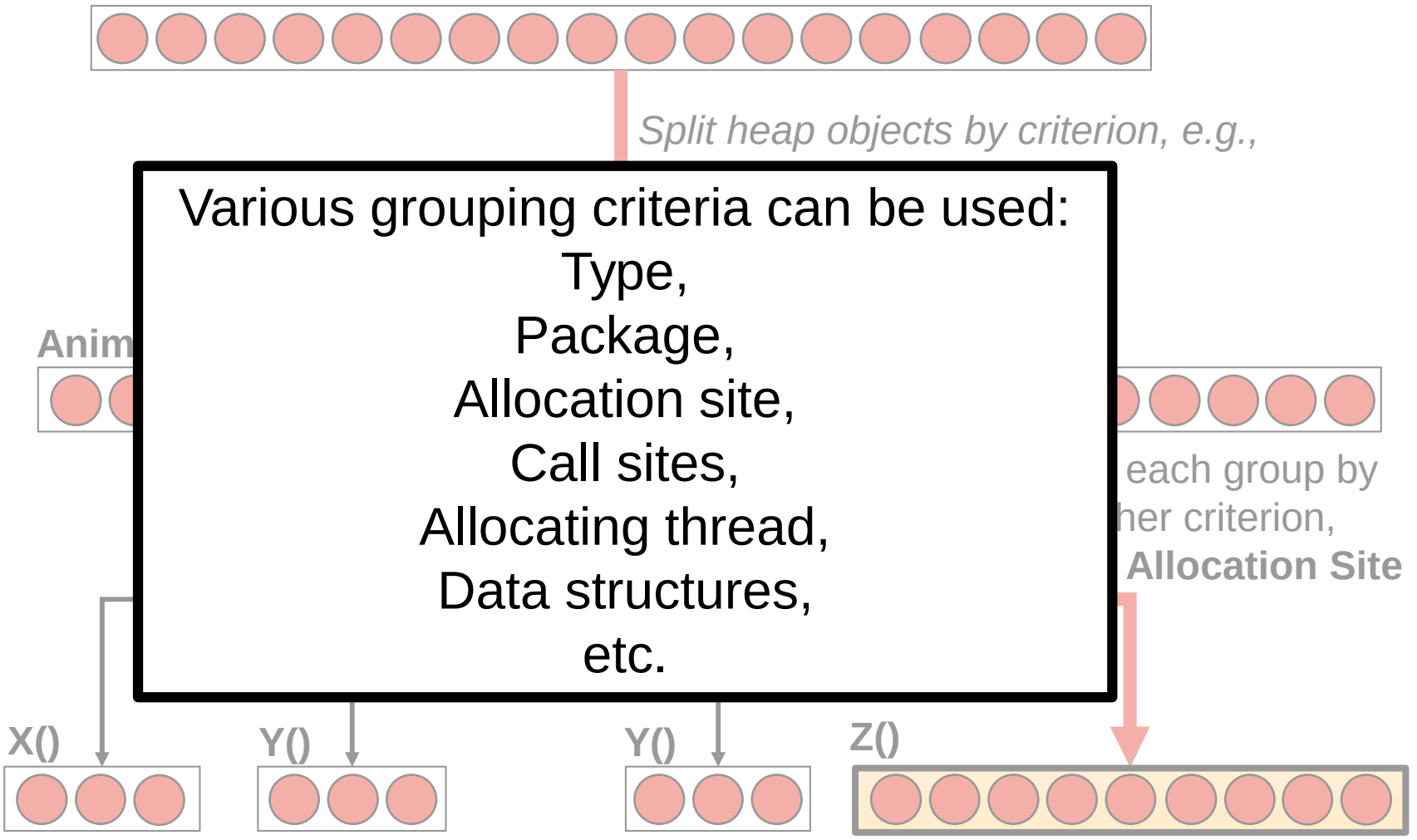
OBJECT CLASSIFICATION



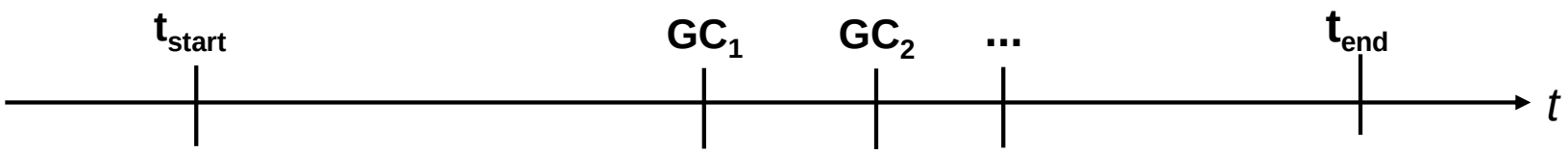
OBJECT CLASSIFICATION



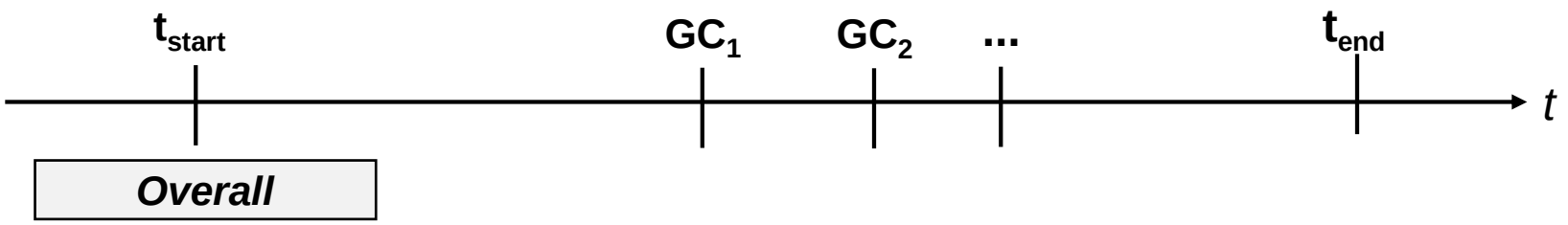
OBJECT CLASSIFICATION



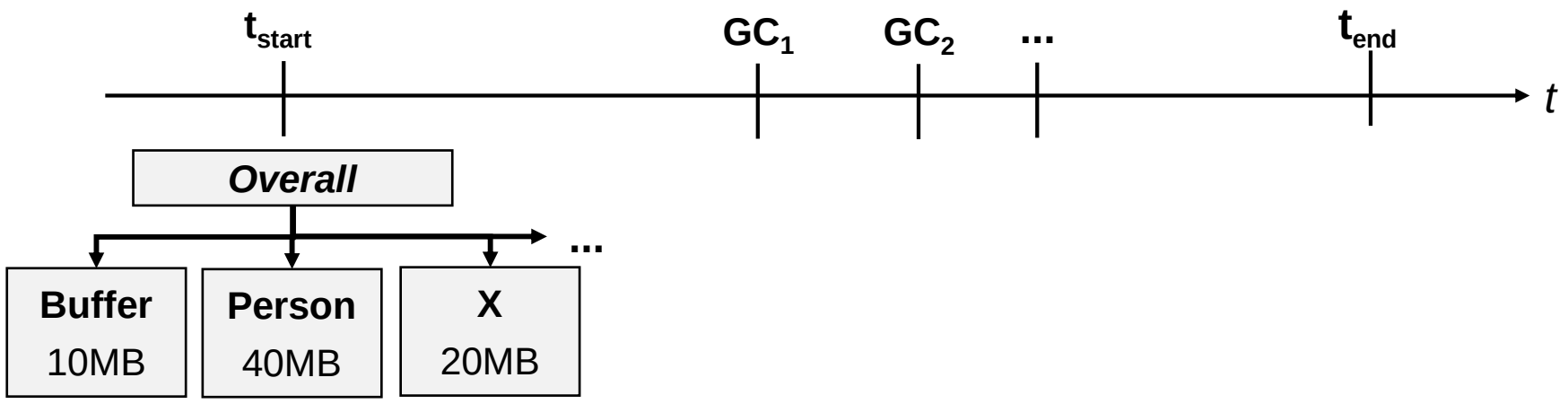
MEMORY EVOLUTION



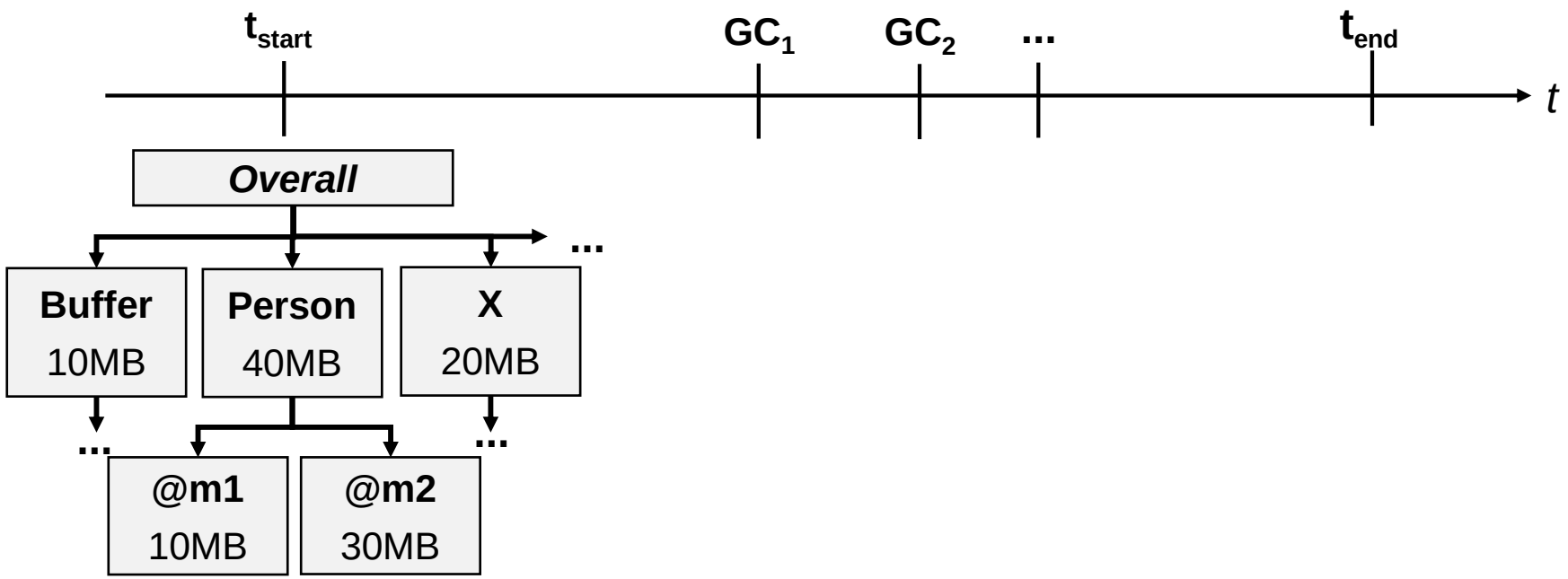
MEMORY EVOLUTION



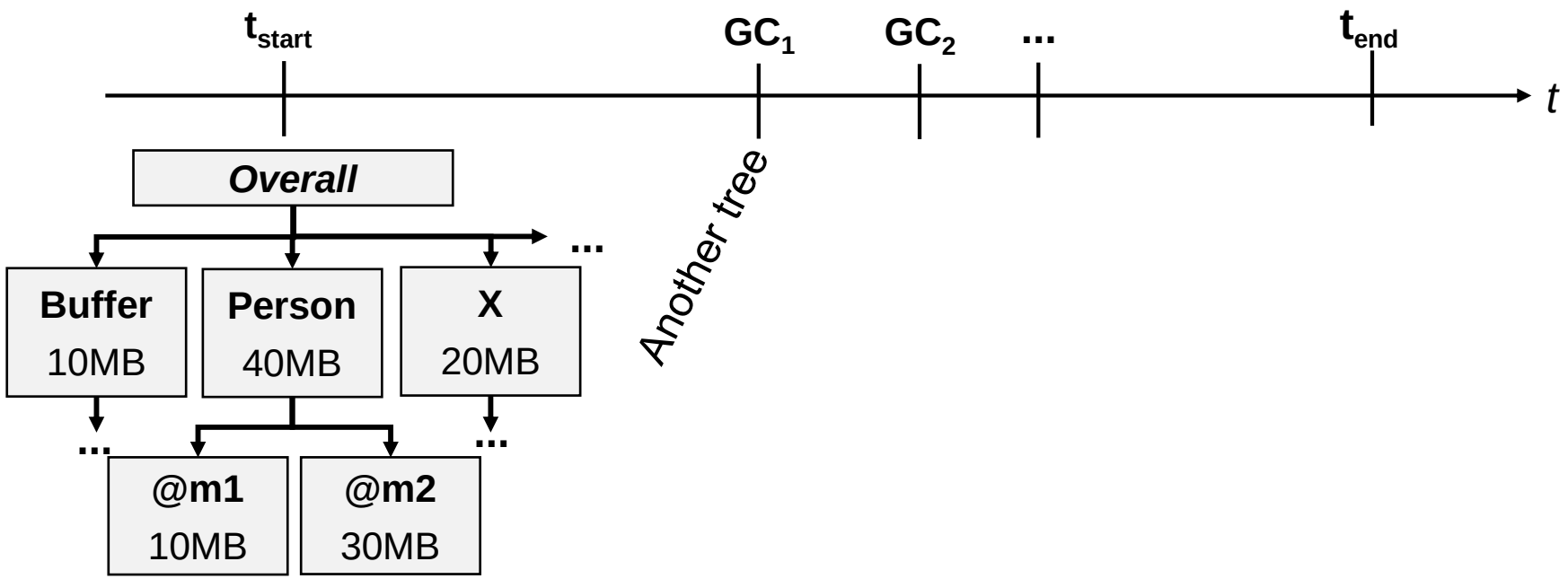
MEMORY EVOLUTION



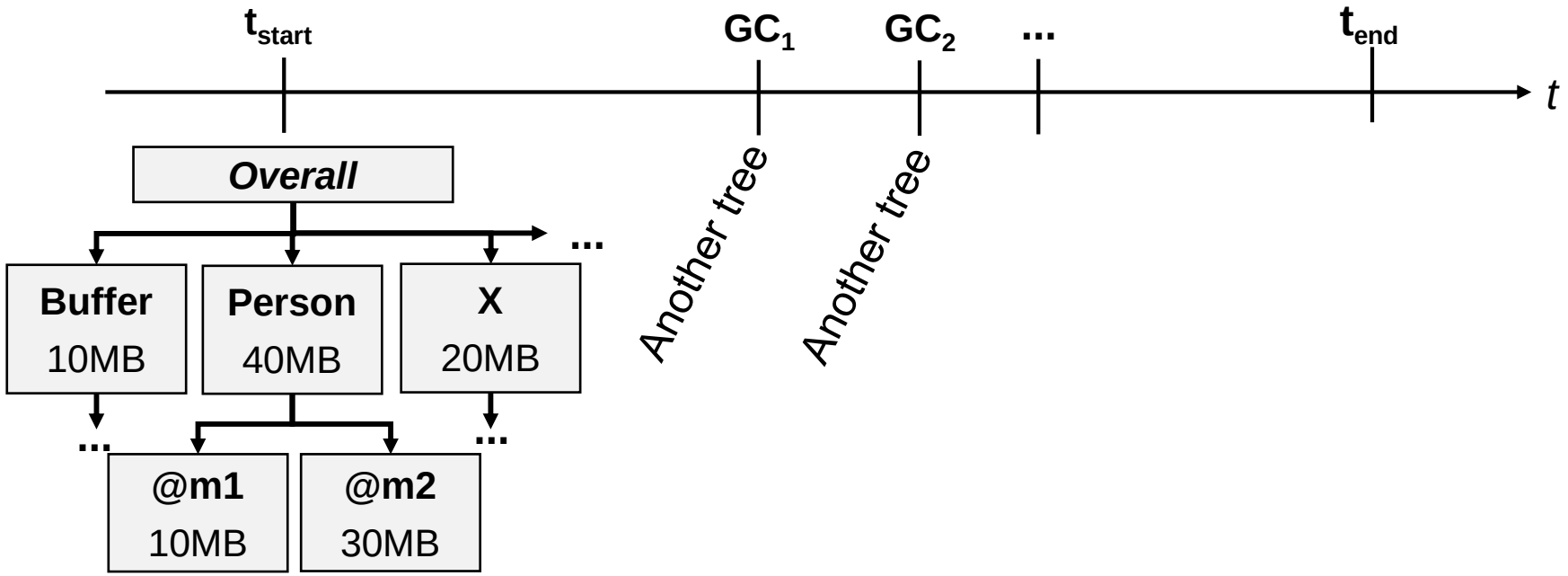
MEMORY EVOLUTION



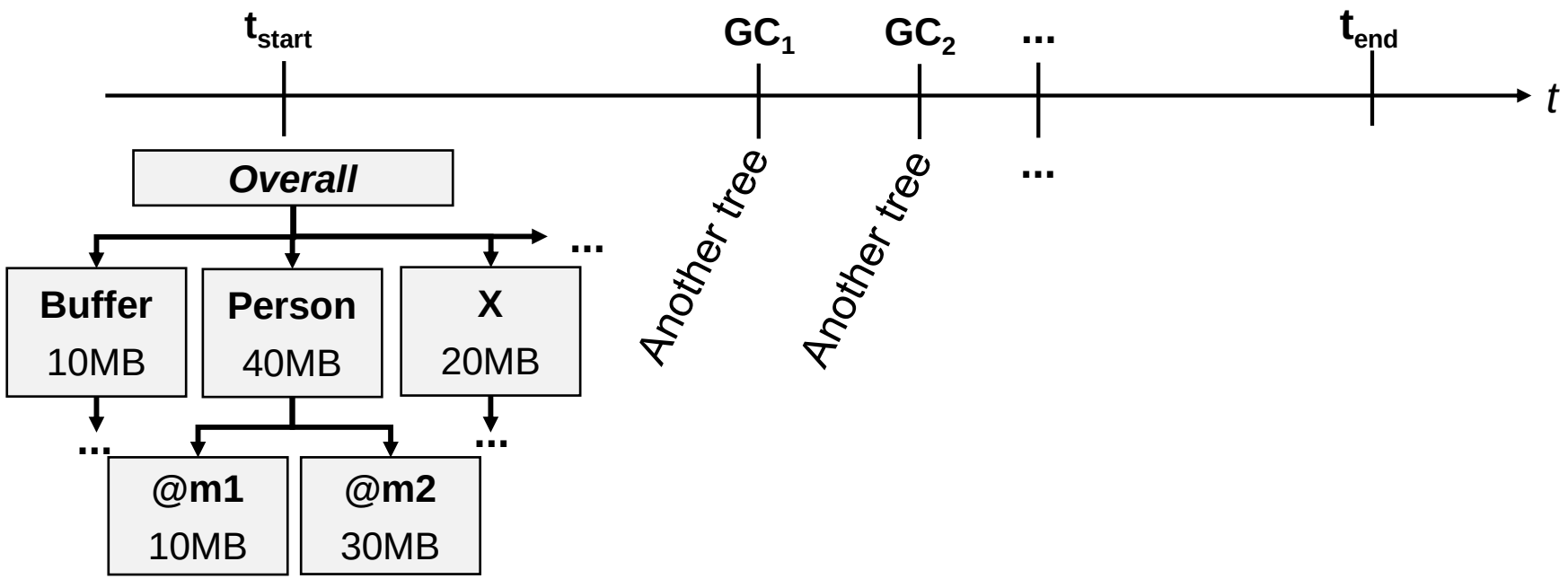
MEMORY EVOLUTION



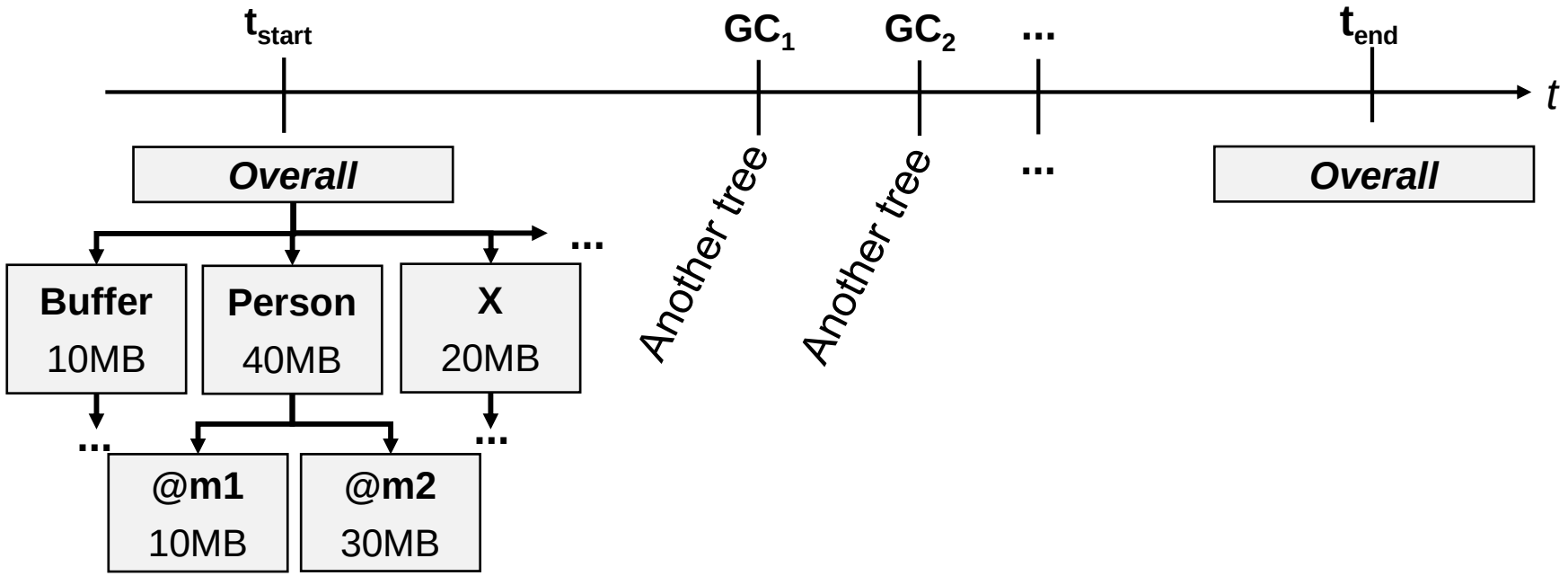
MEMORY EVOLUTION



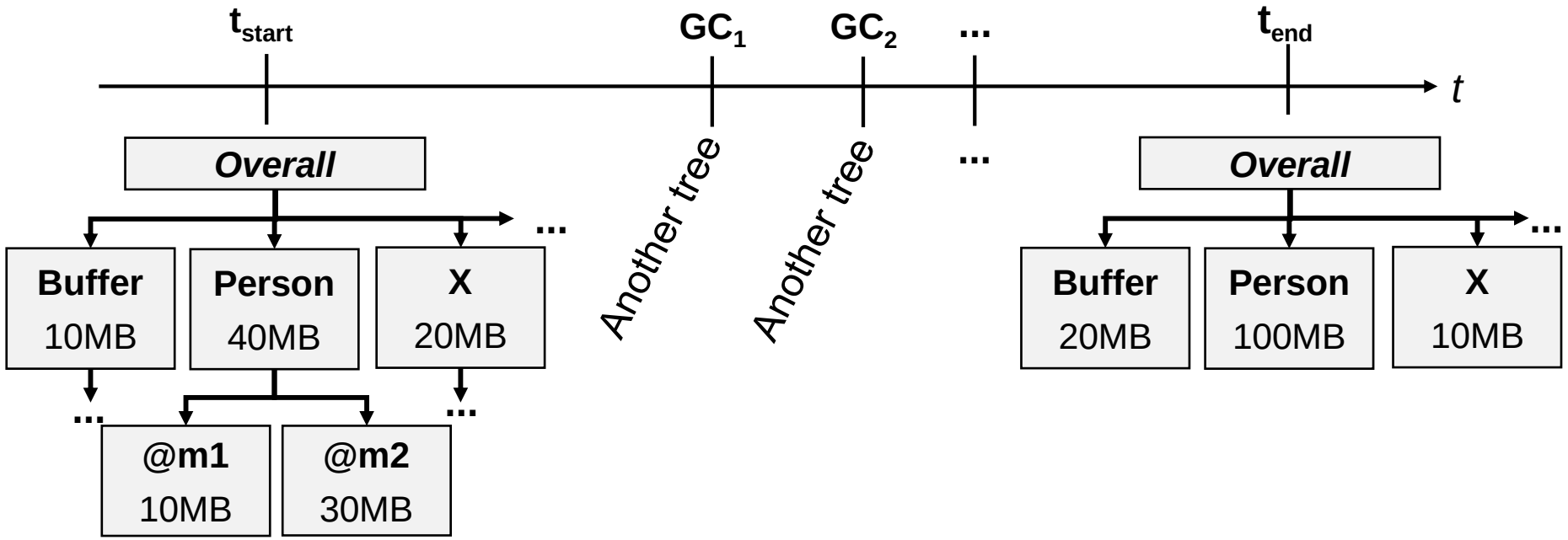
MEMORY EVOLUTION



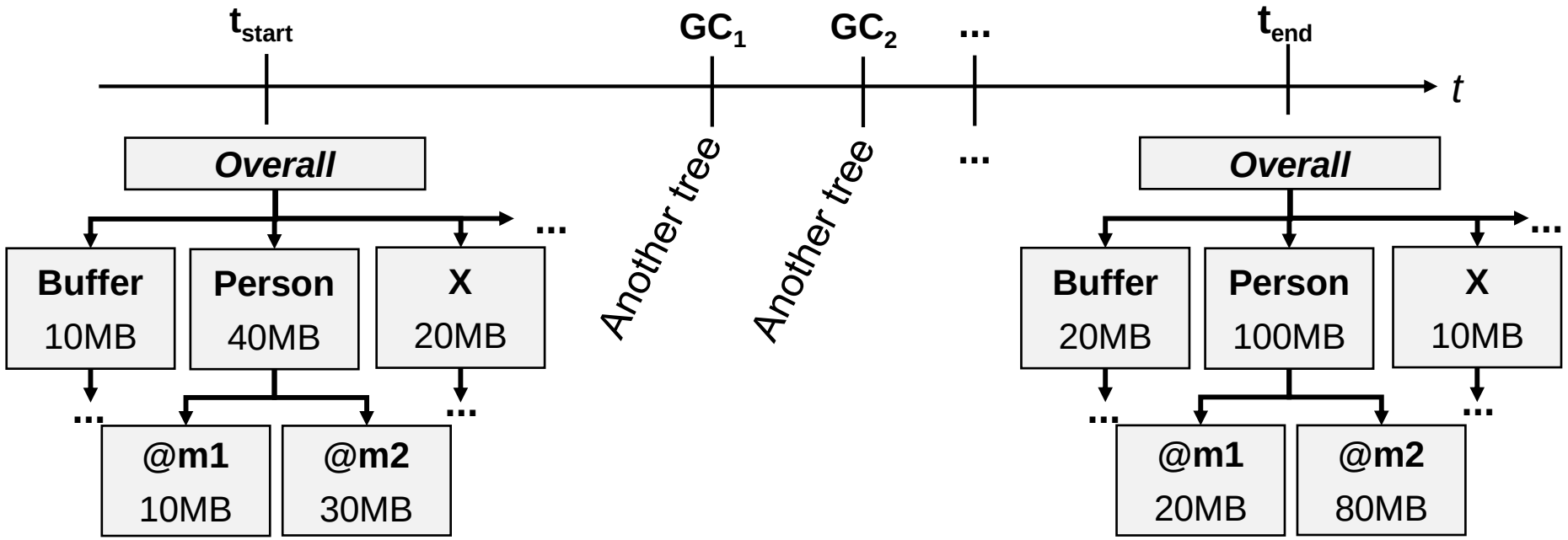
MEMORY EVOLUTION



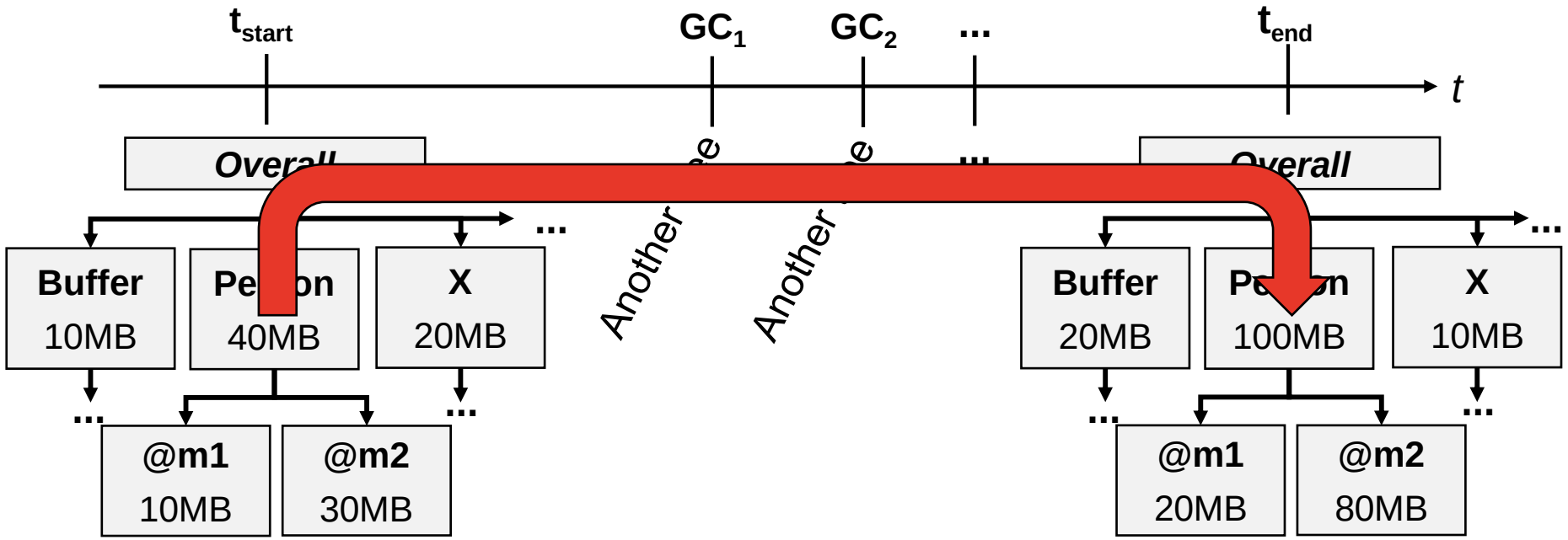
MEMORY EVOLUTION



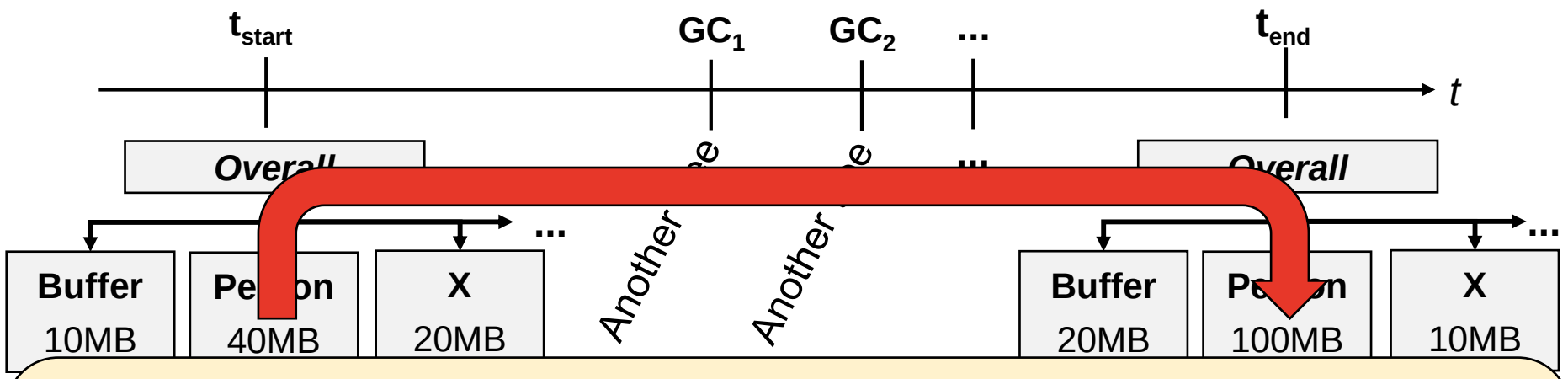
MEMORY EVOLUTION



MEMORY EVOLUTION



MEMORY EVOLUTION



Question: How to visualize memory trees and their evolution?

REQUIREMENTS

REQUIREMENTS

“Bigger is more” principle

REQUIREMENTS

“Bigger is more” principle

Suitable for interactive visualization

REQUIREMENTS

“Bigger is more” principle

Suitable for interactive visualization

Proven and easy to understand

REQUIREMENTS

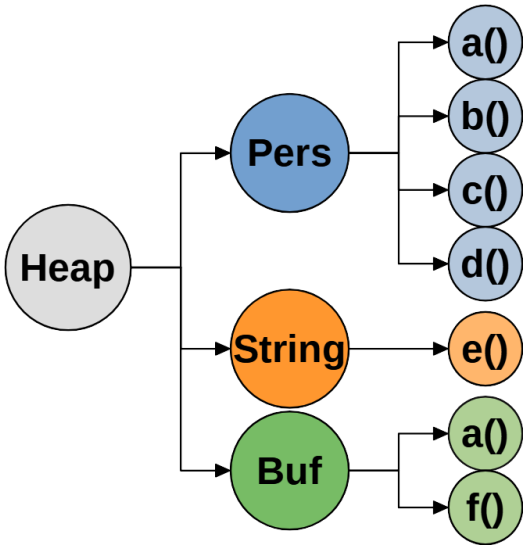
“Bigger is more” principle

Suitable for interactive visualization

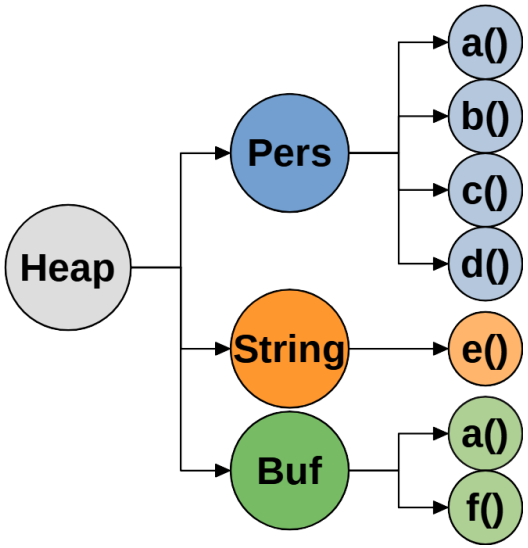
Proven and easy to understand

Stable layout on evolving data

VISUALIZATIONS COMPARED

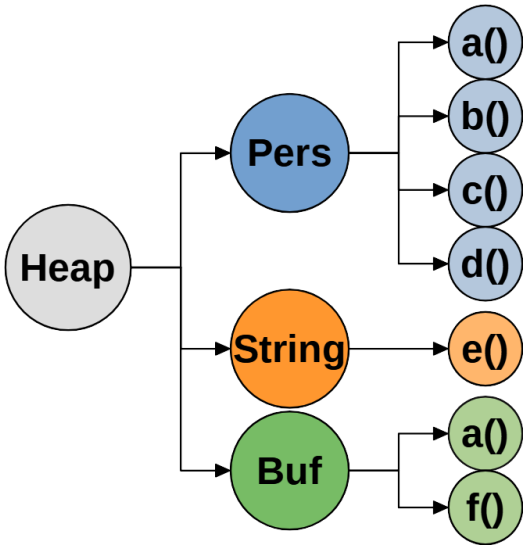


VISUALIZATIONS COMPARED

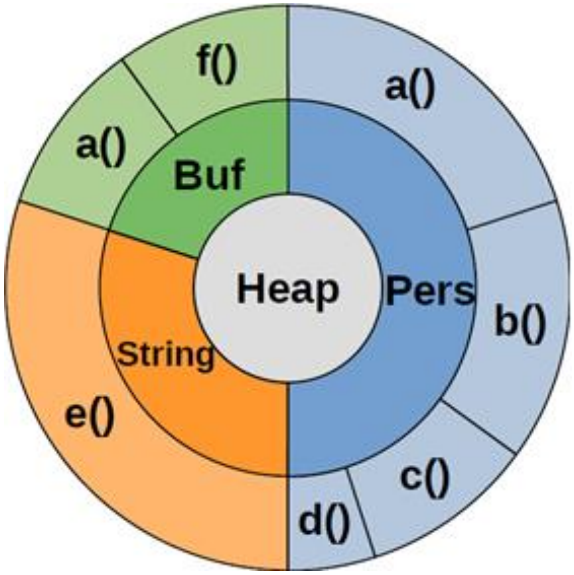


Node-Link Diag.

VISUALIZATIONS COMPARED

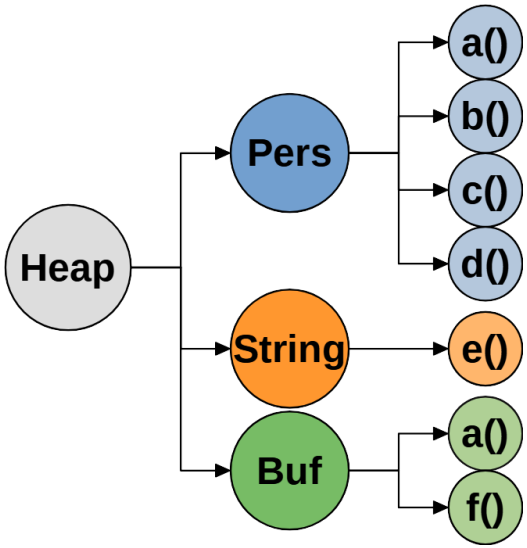


Node-Link Diag.

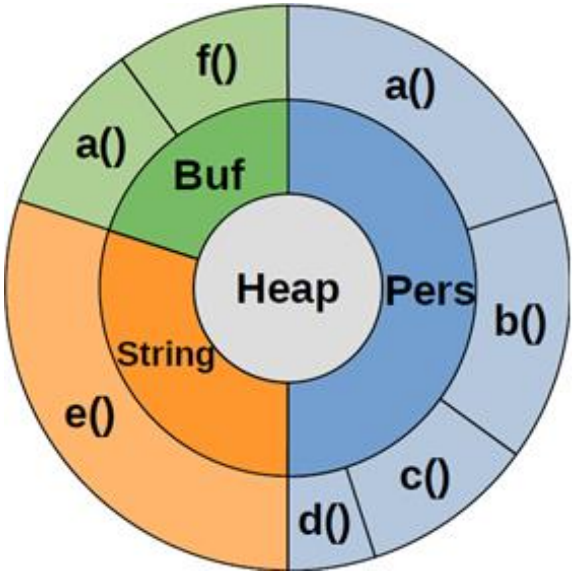


Sunburst

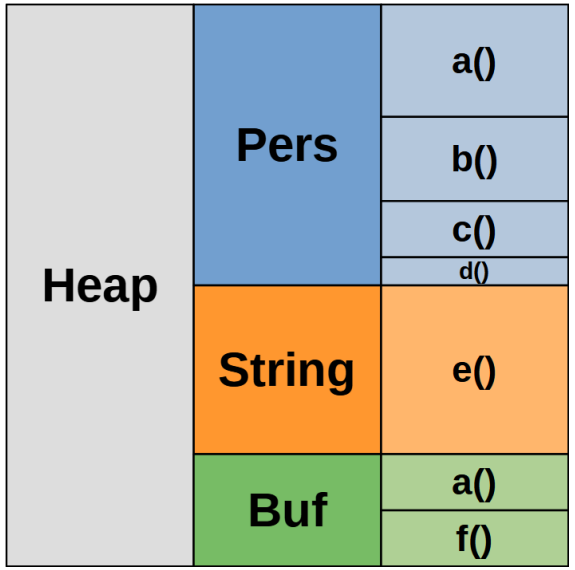
VISUALIZATIONS COMPARED



Node-Link Diag.



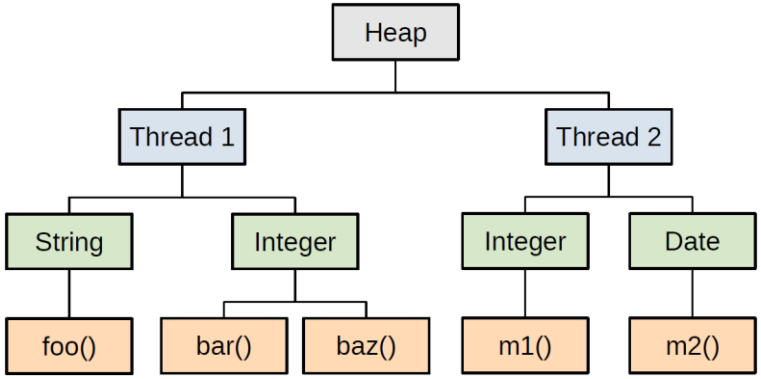
Sunburst



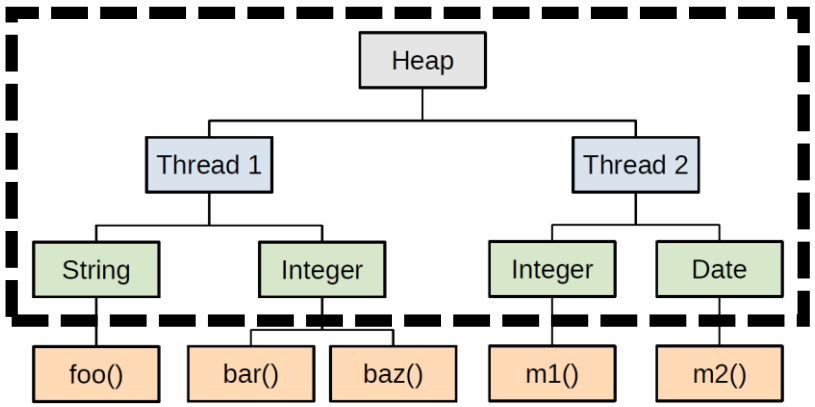
Icicle

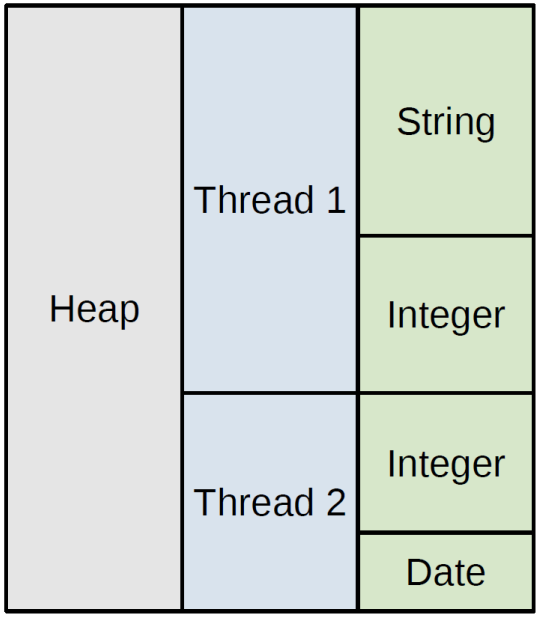
HEAP STATE VISUALIZATION

HEAP STATE VISUALIZATION

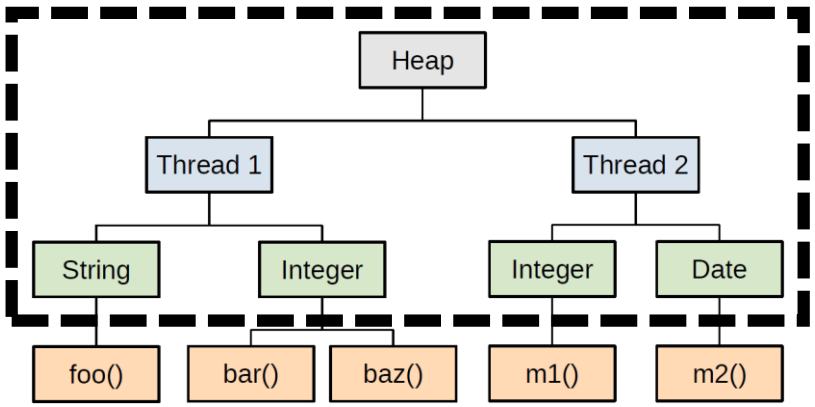


HEAP STATE VISUALIZATION

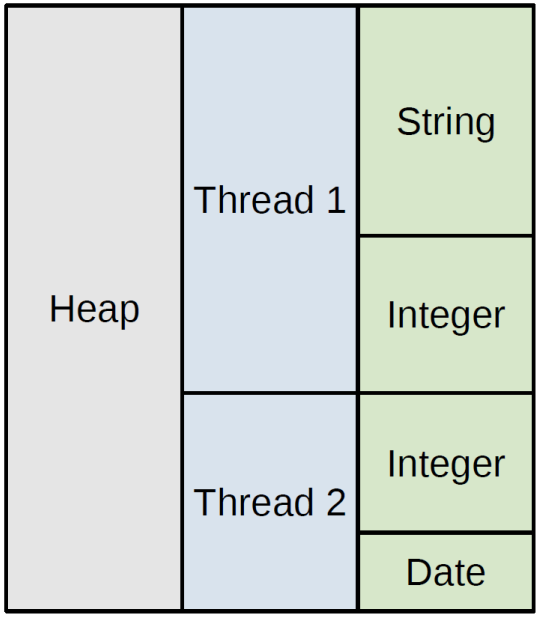




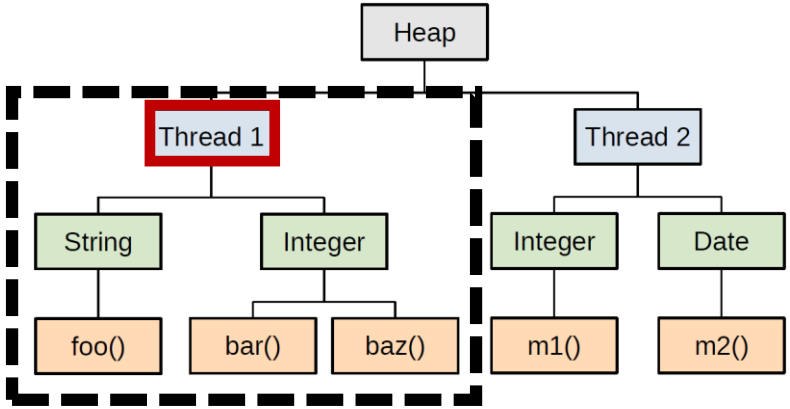
HEAP STATE VISUALIZATION



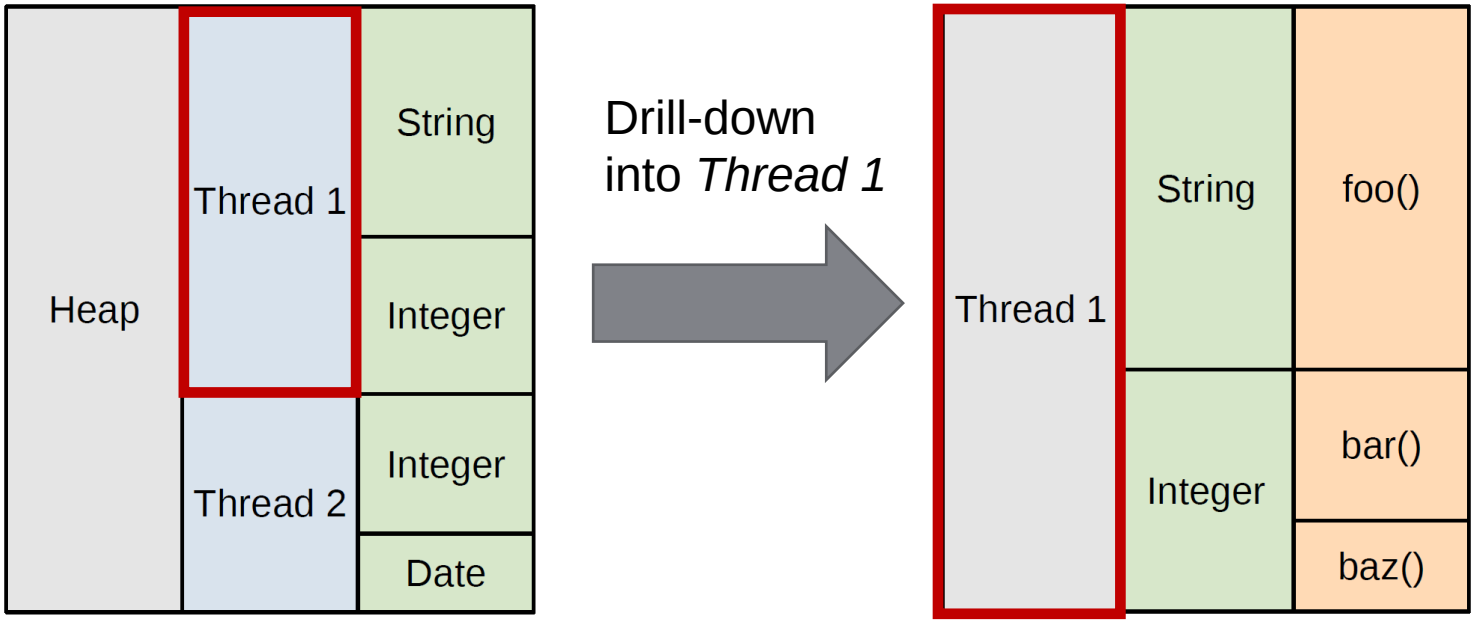
Drill-down



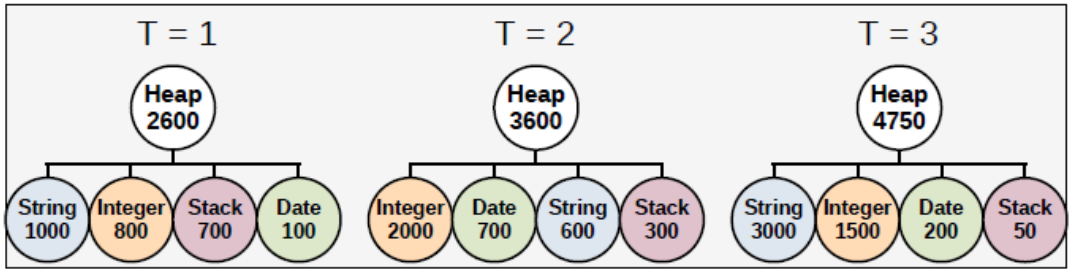
HEAP STATE VISUALIZATION



Drill-down

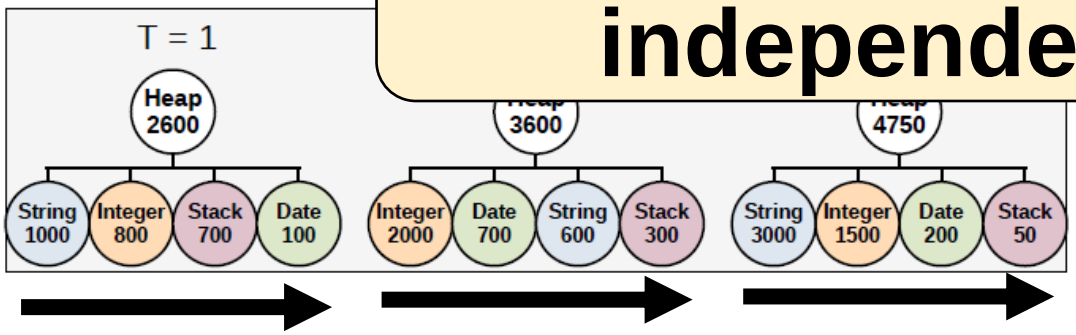


HEAP EVOLUTION VISUALIZATION

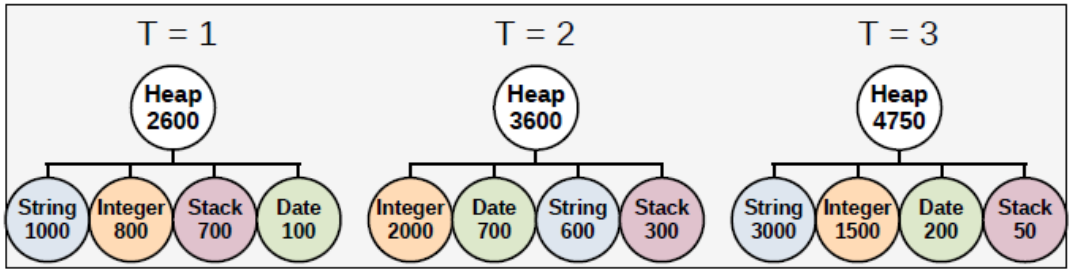


HEAP EVOLUTION VISUALIZATION

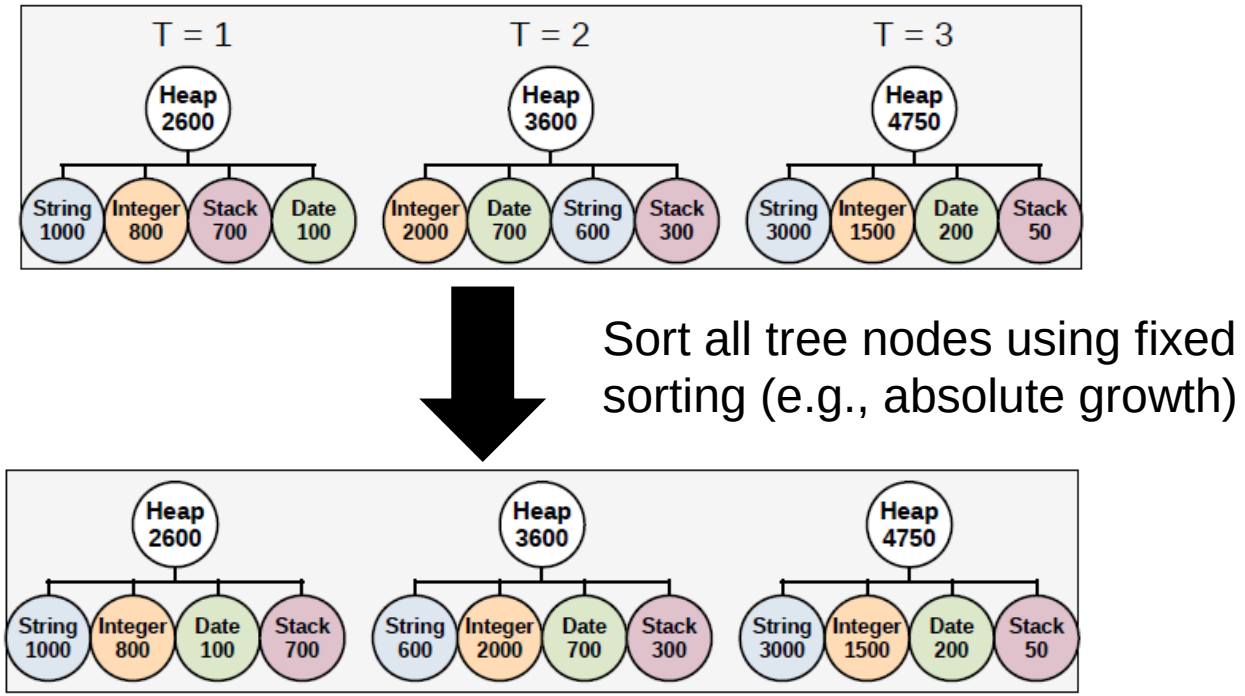
Each tree sorted independently



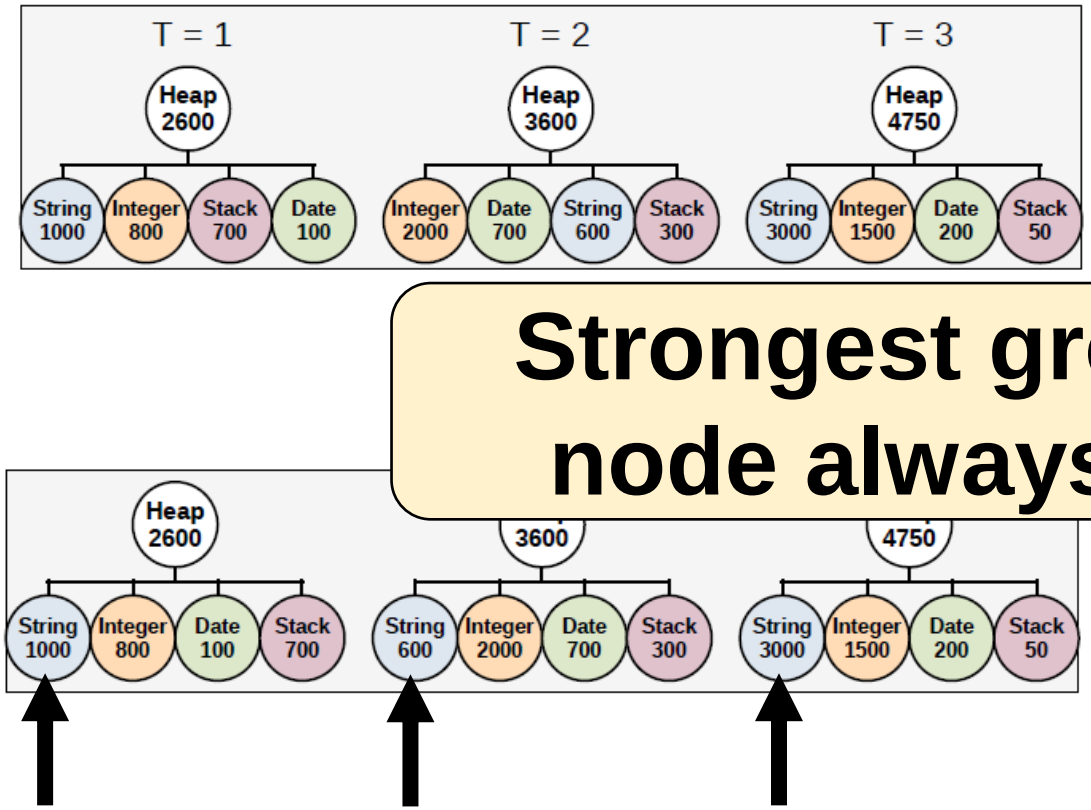
HEAP EVOLUTION VISUALIZATION



HEAP EVOLUTION VISUALIZATION

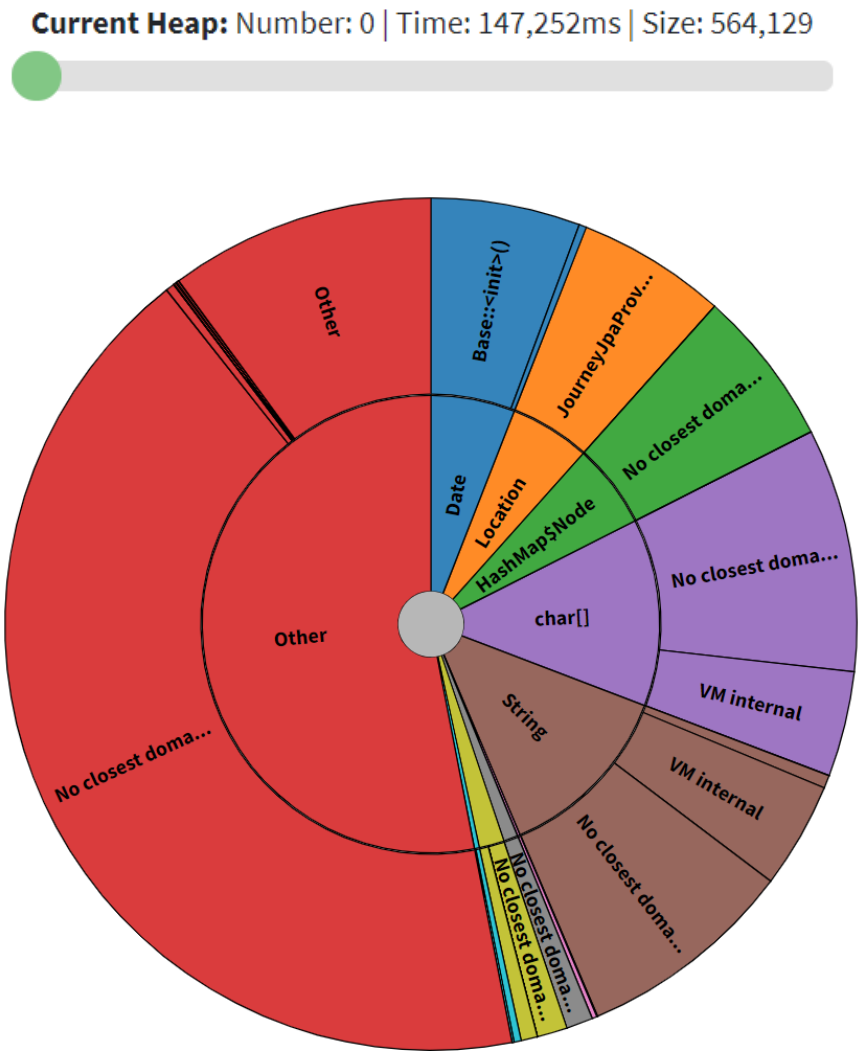


HEAP EVOLUTION VISUALIZATION



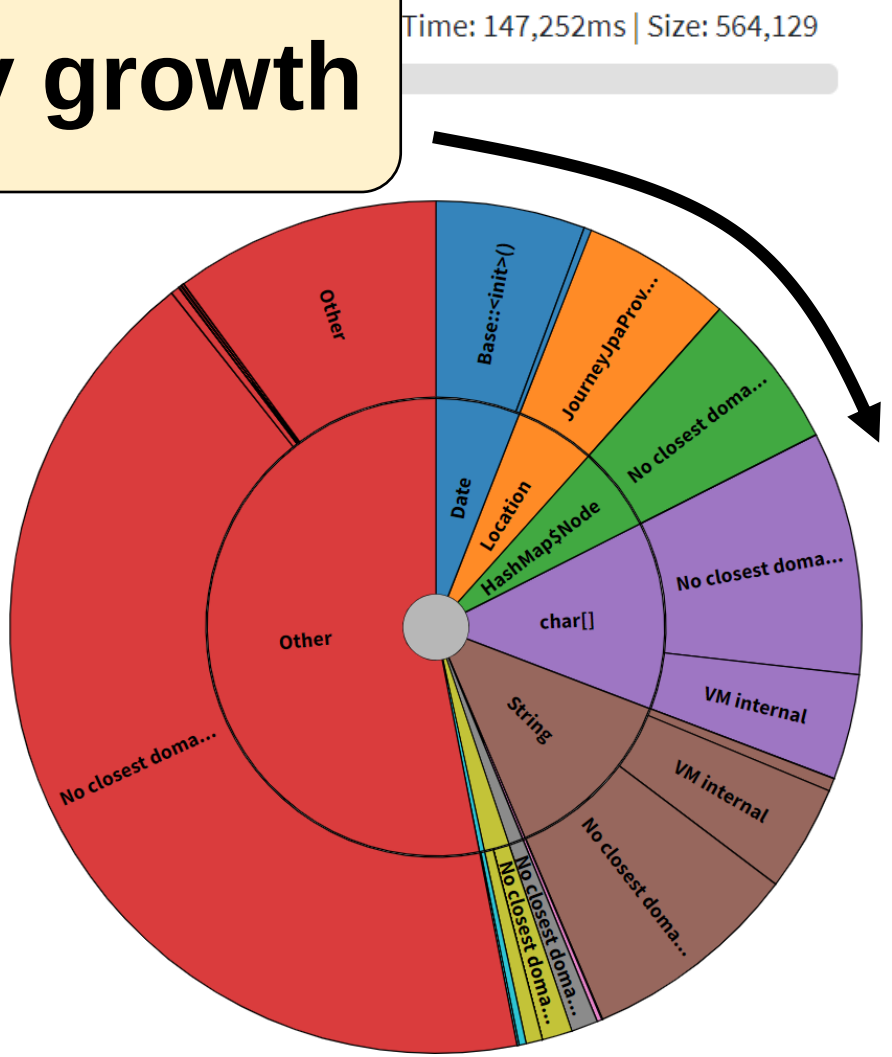
RELATIVE POSITION ANIMATION

RELATIVE POSITION ANIMATION



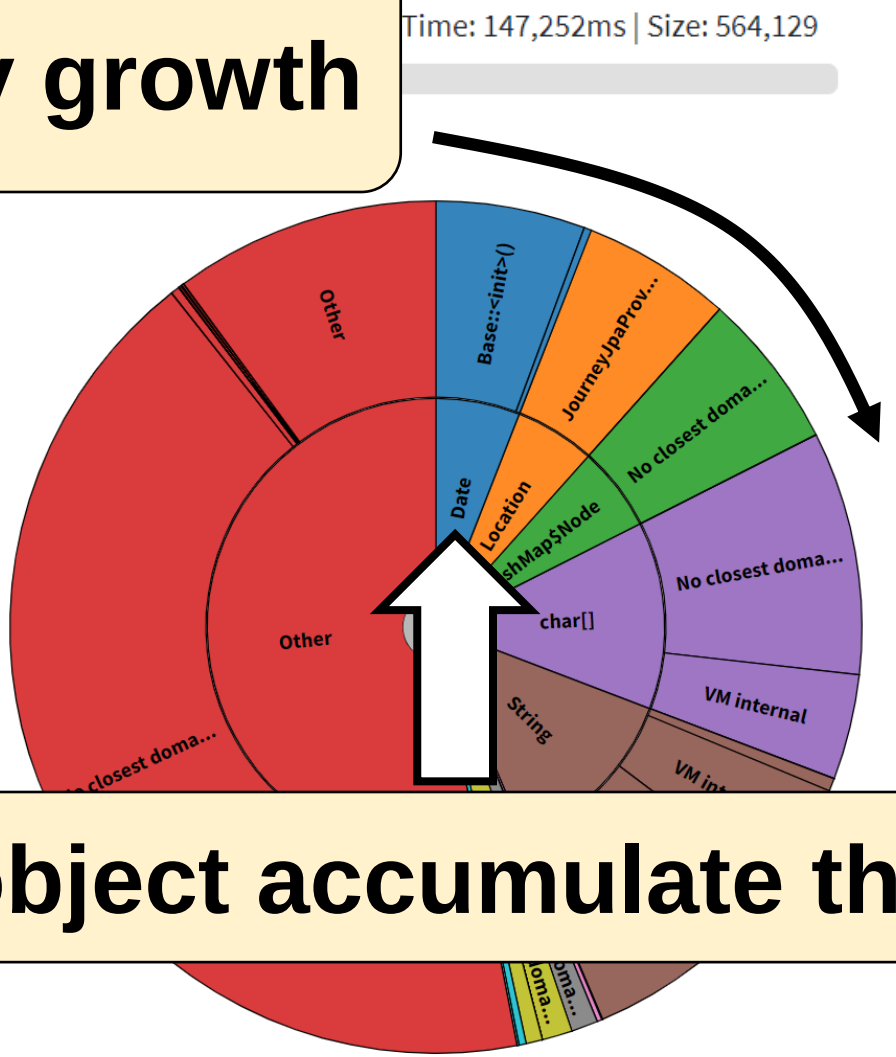
RELATIVE POSITION ANIMATION

Sorted by growth



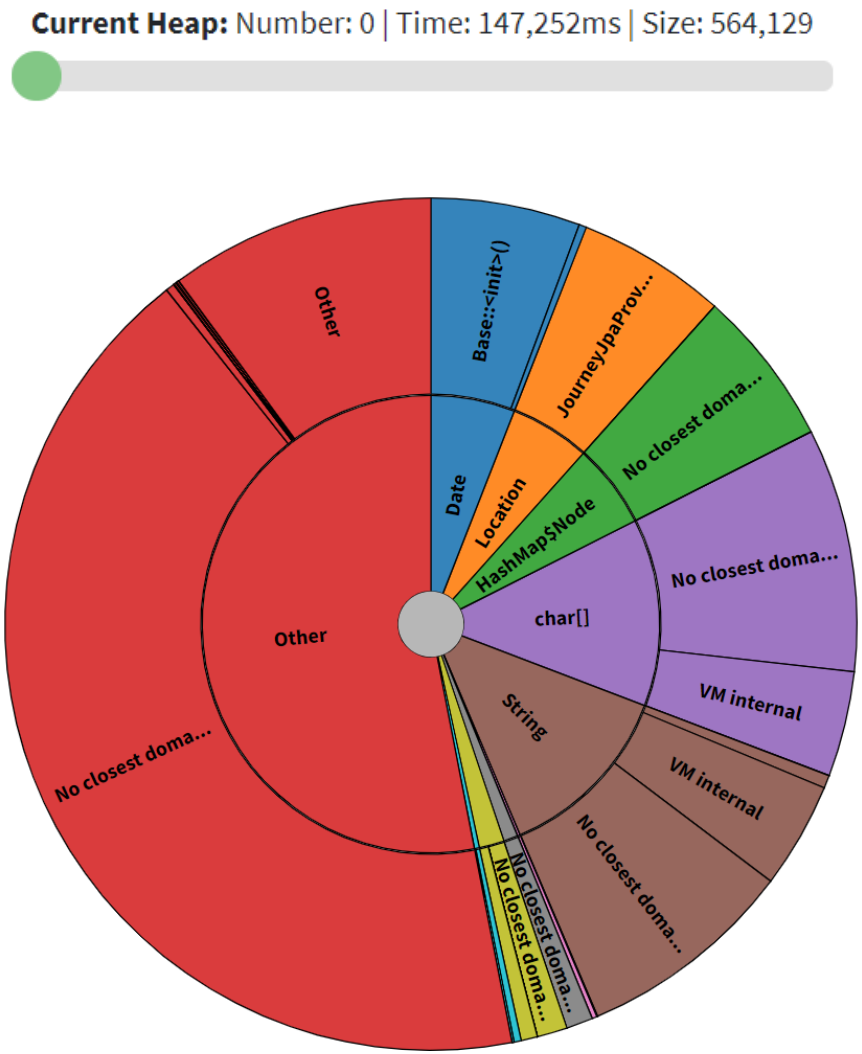
RELATIVE POSITION ANIMATION

Sorted by growth

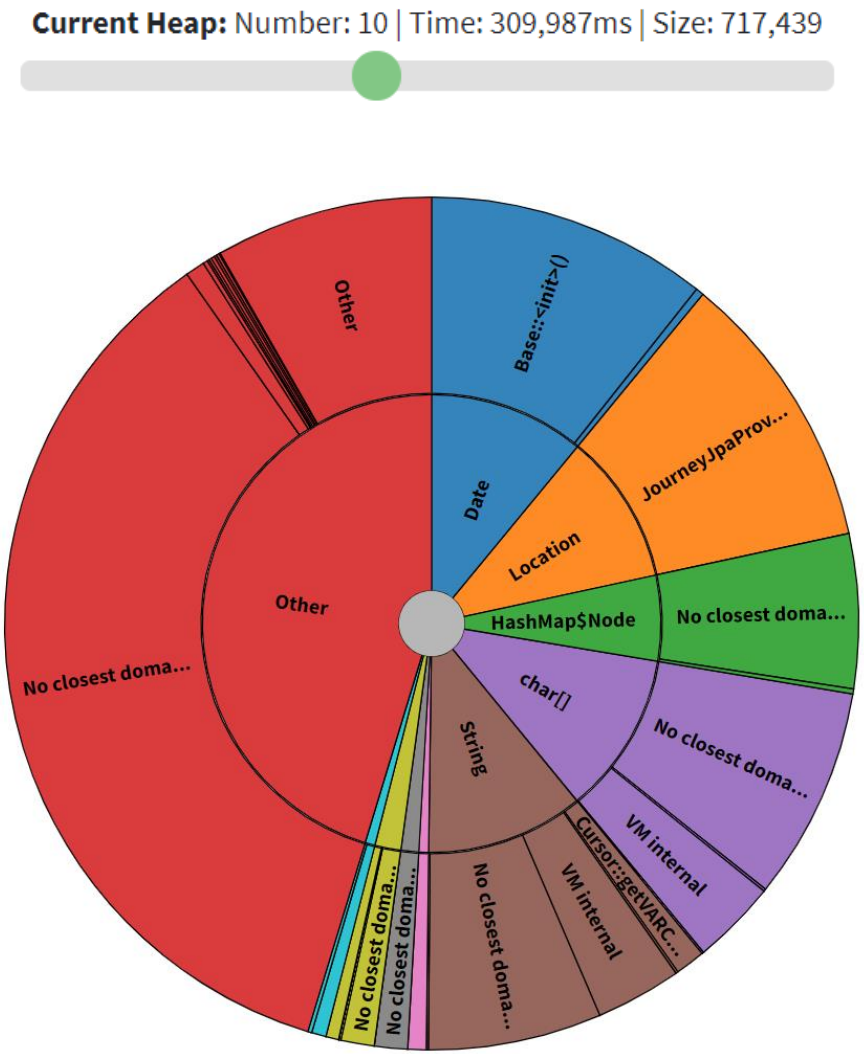


***Date* object accumulate the most**

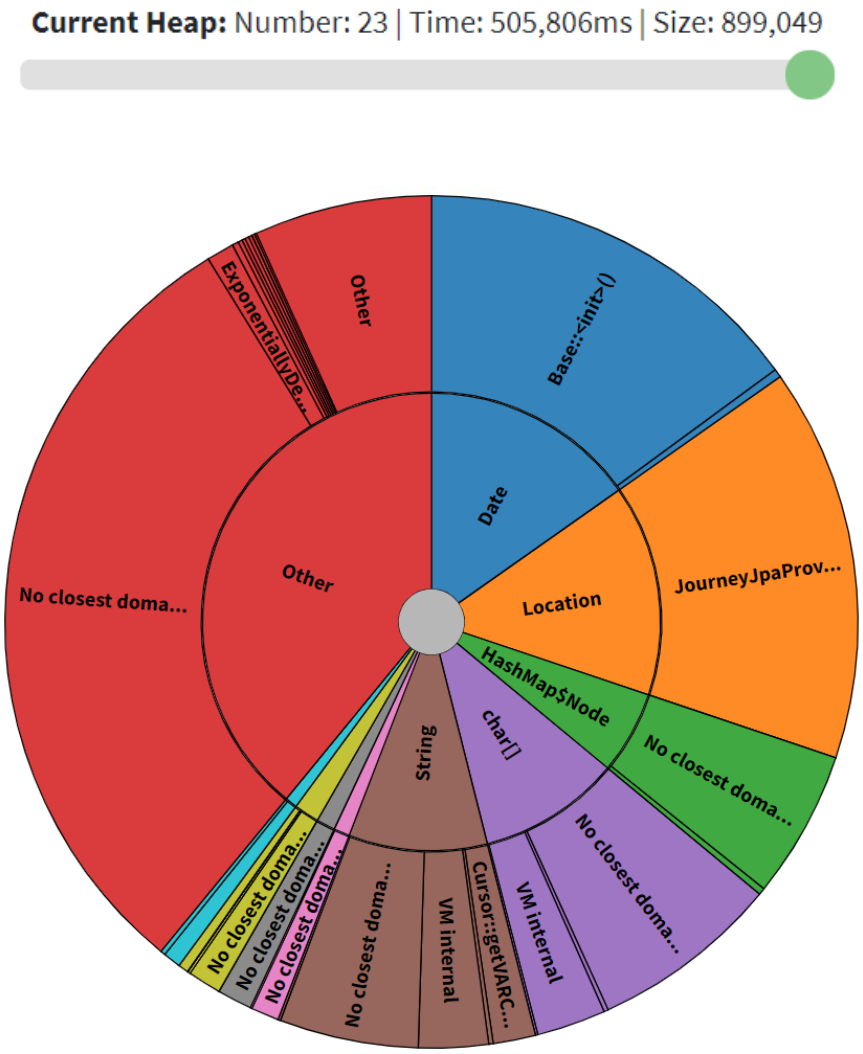
RELATIVE POSITION ANIMATION



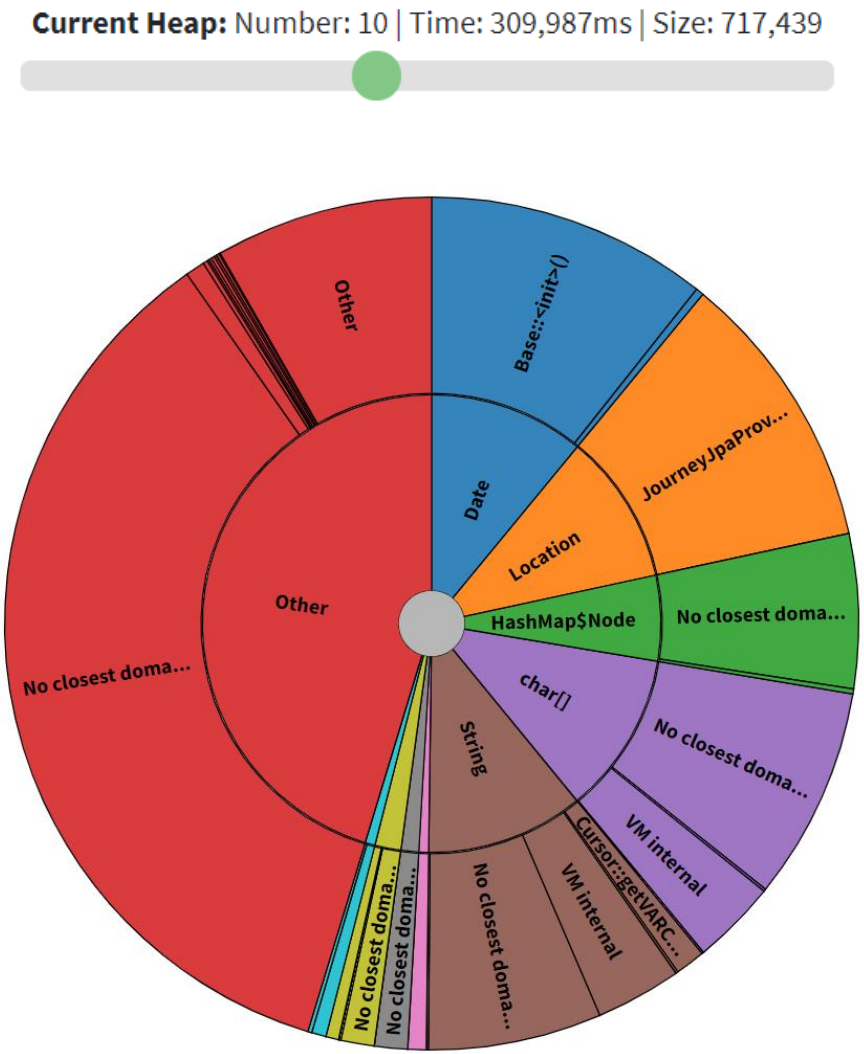
RELATIVE POSITION ANIMATION



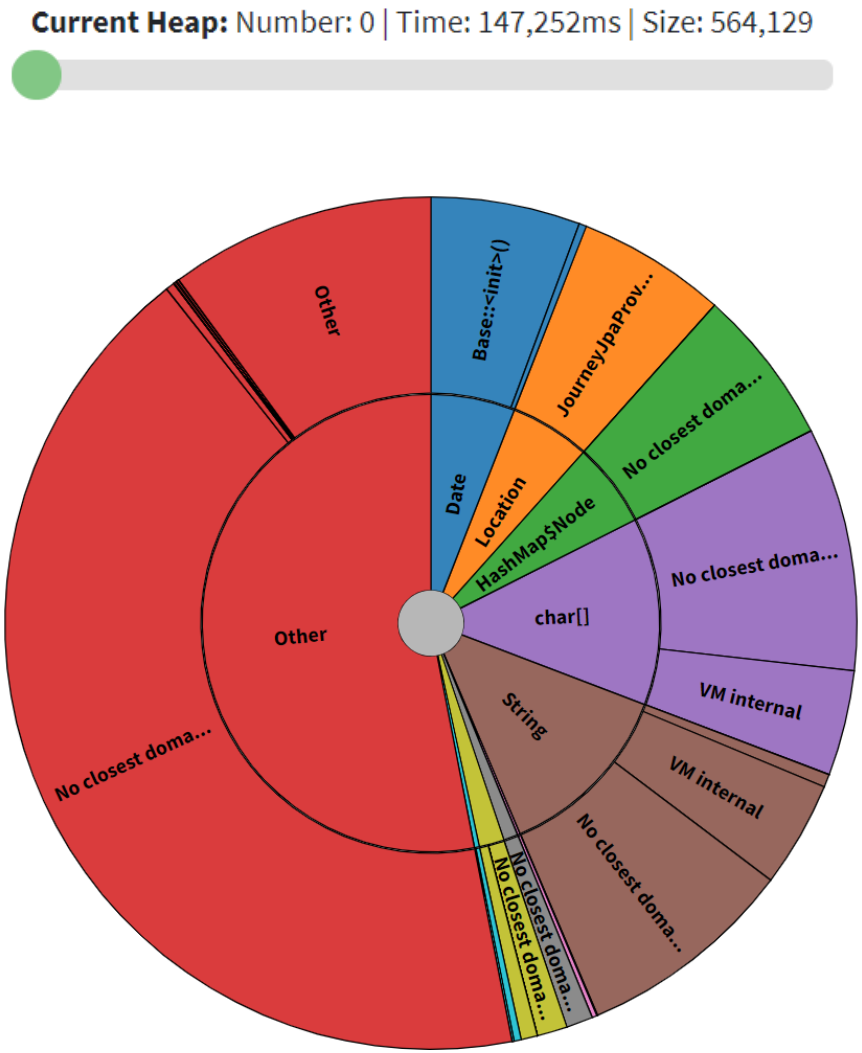
RELATIVE POSITION ANIMATION



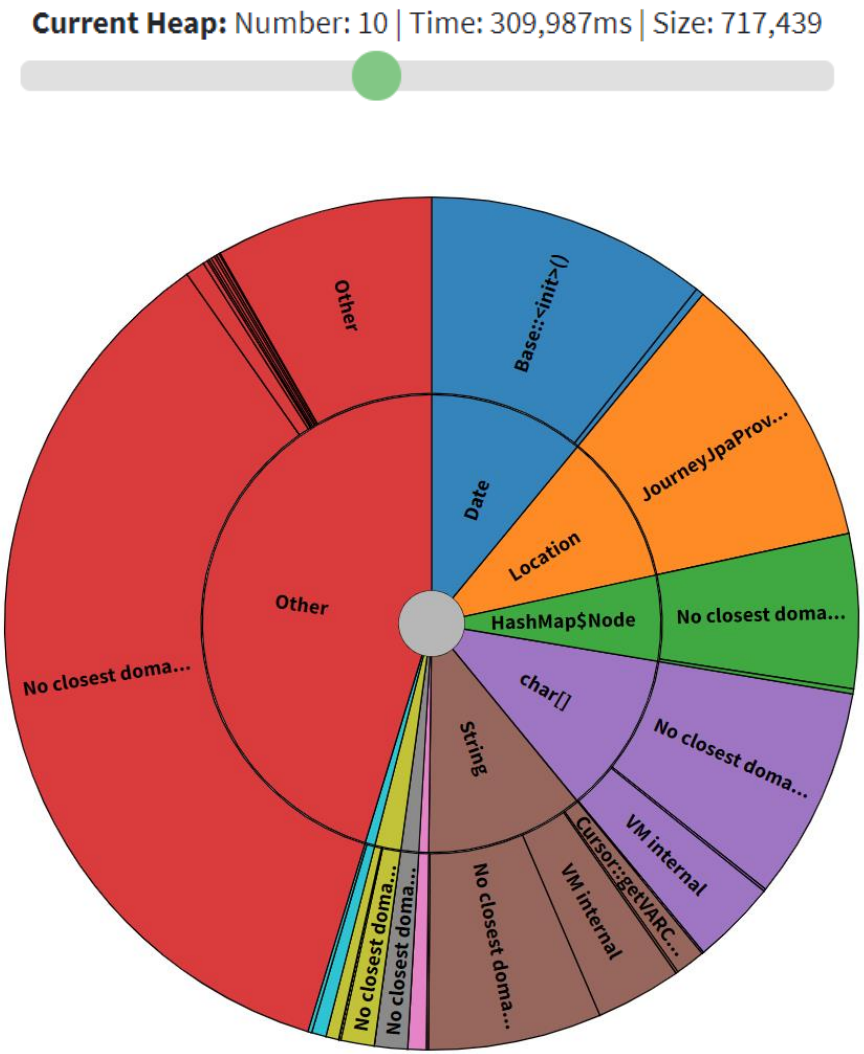
RELATIVE POSITION ANIMATION



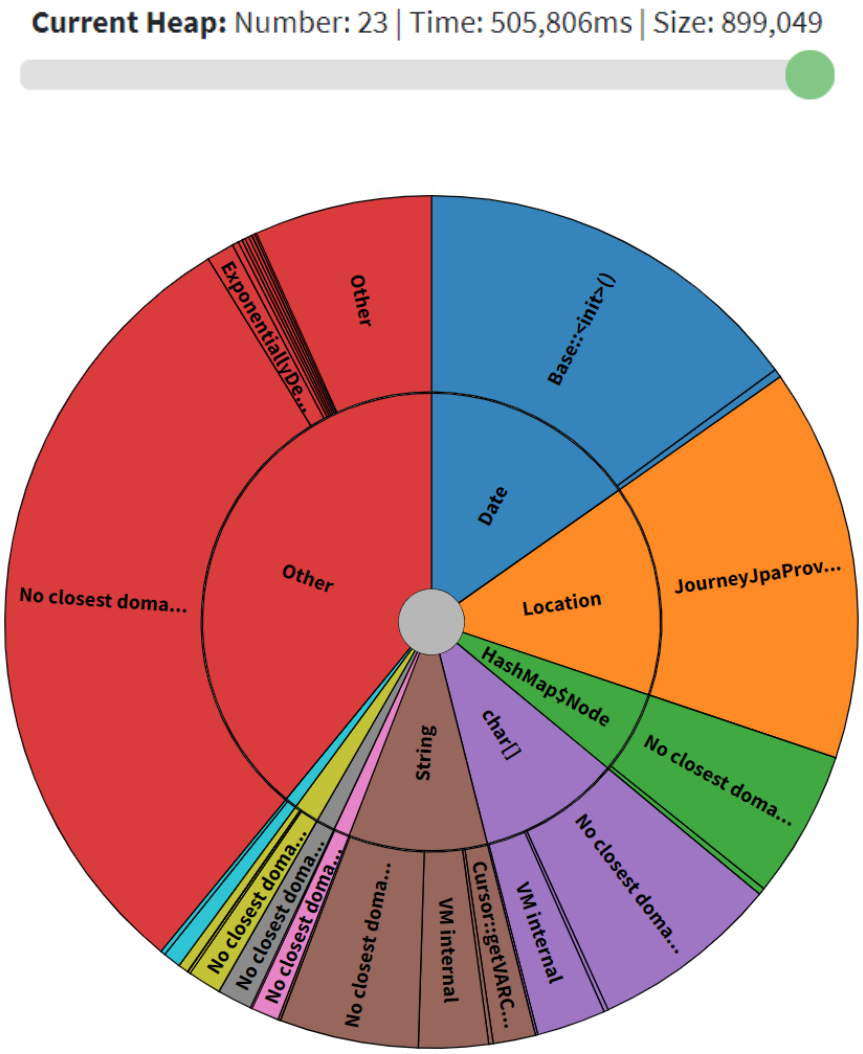
RELATIVE POSITION ANIMATION



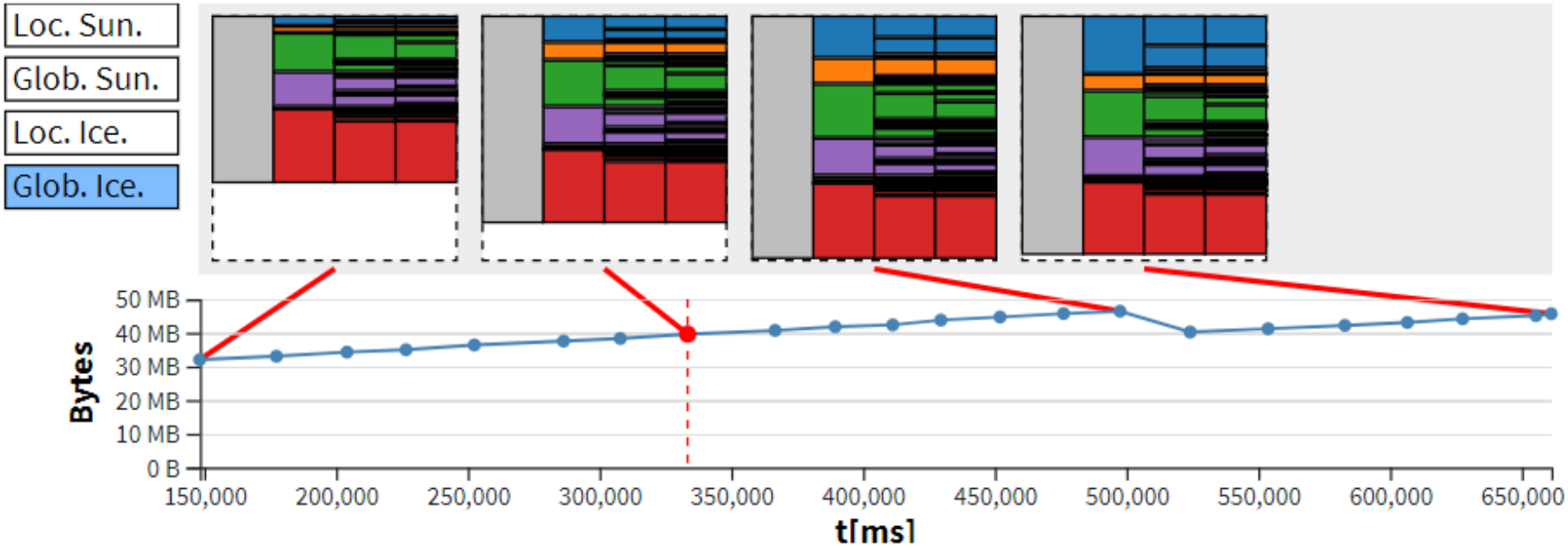
RELATIVE POSITION ANIMATION



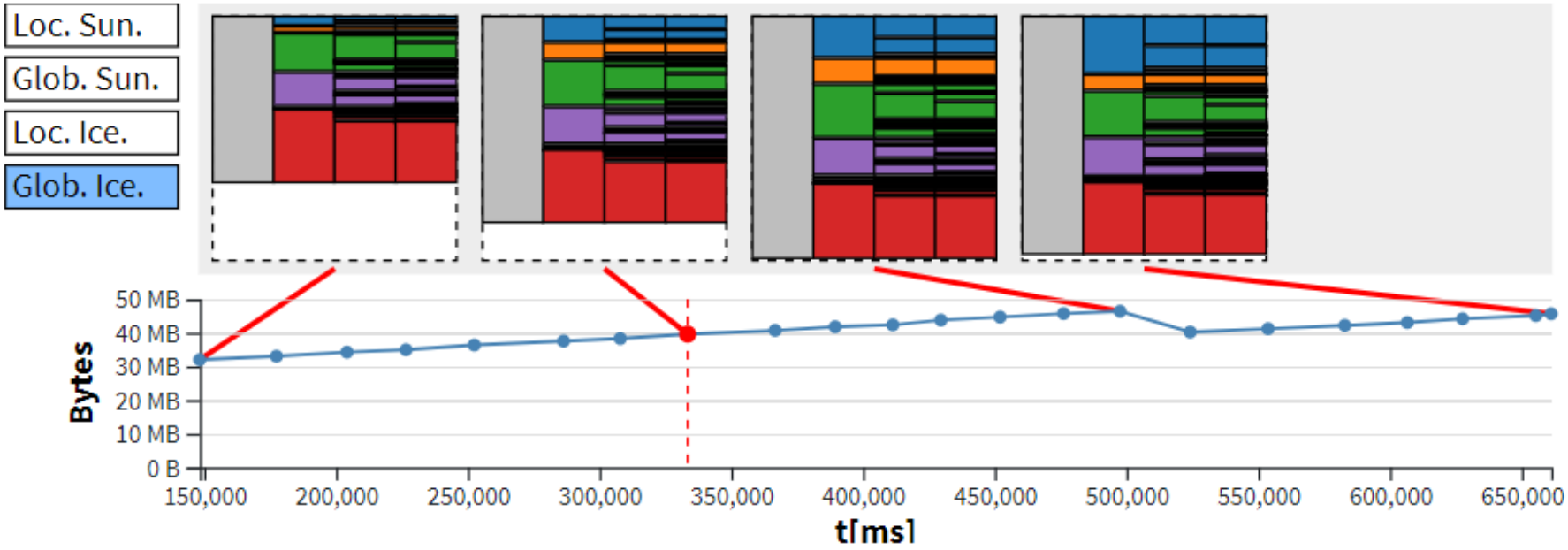
RELATIVE POSITION ANIMATION



TIMELINE VIEW

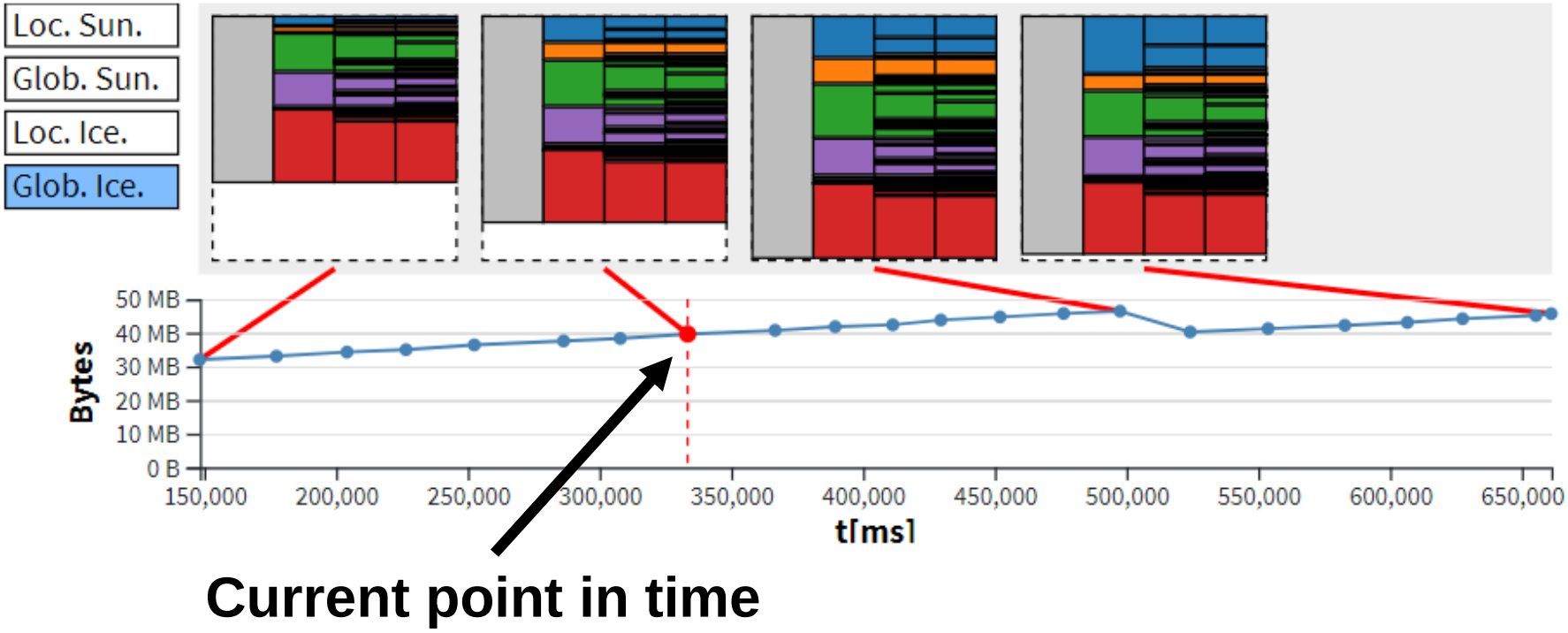


TIMELINE VIEW



Small multiples at different points in time

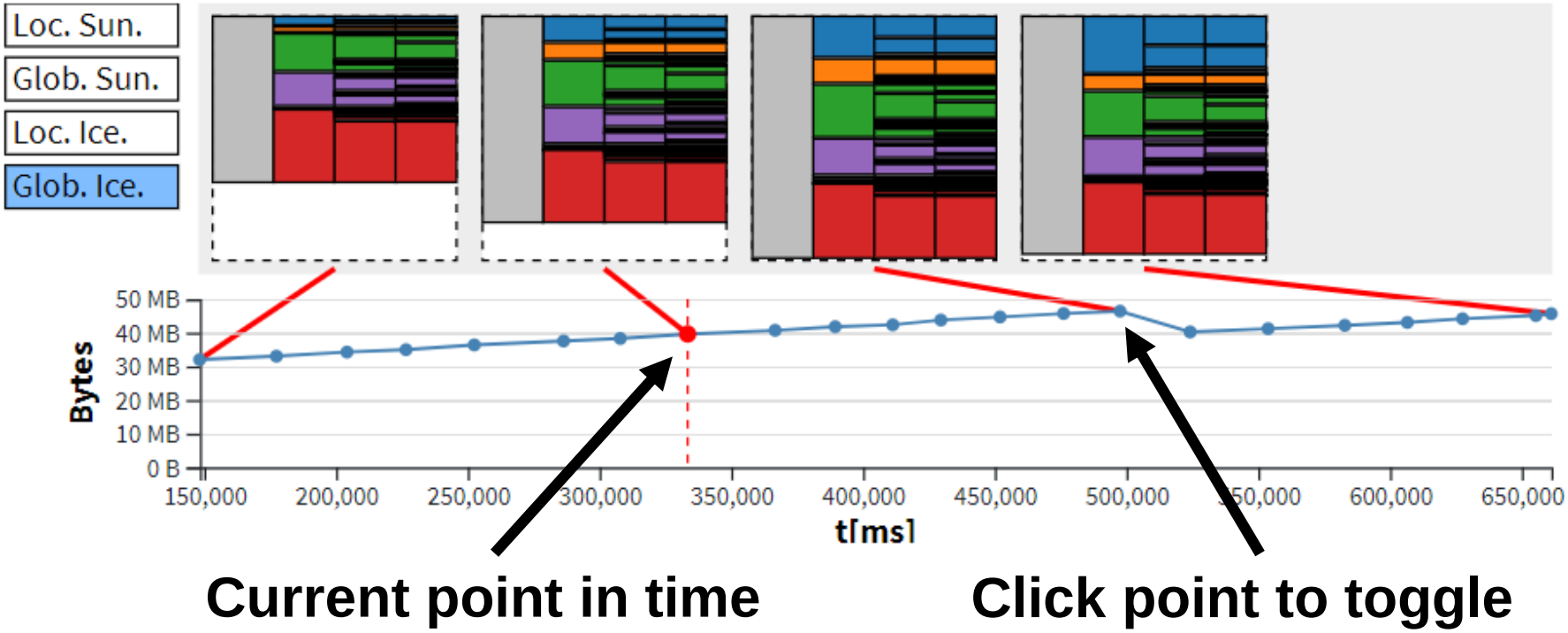
TIMELINE VIEW



Current point in time

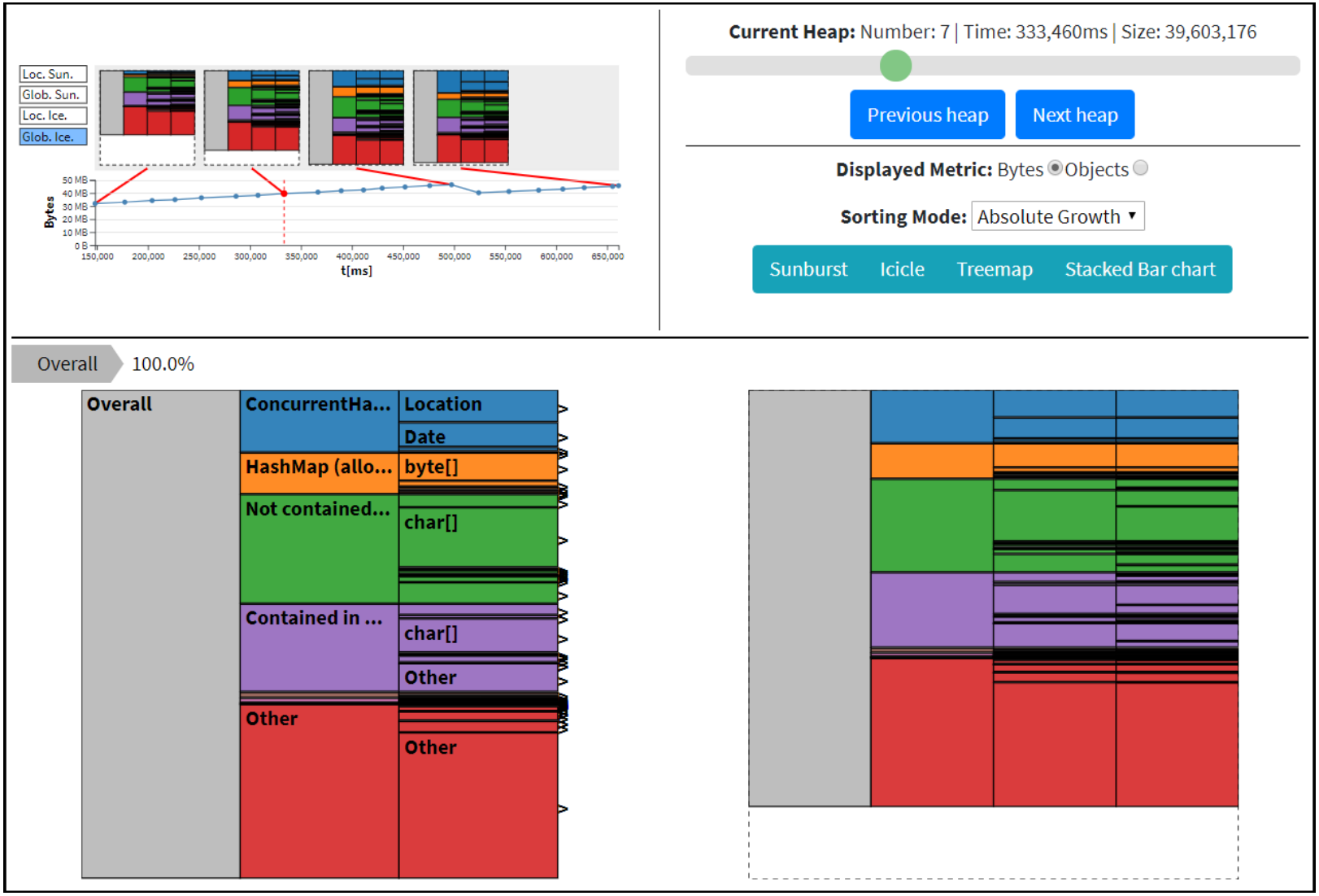
Small multiples at different points in time

TIMELINE VIEW

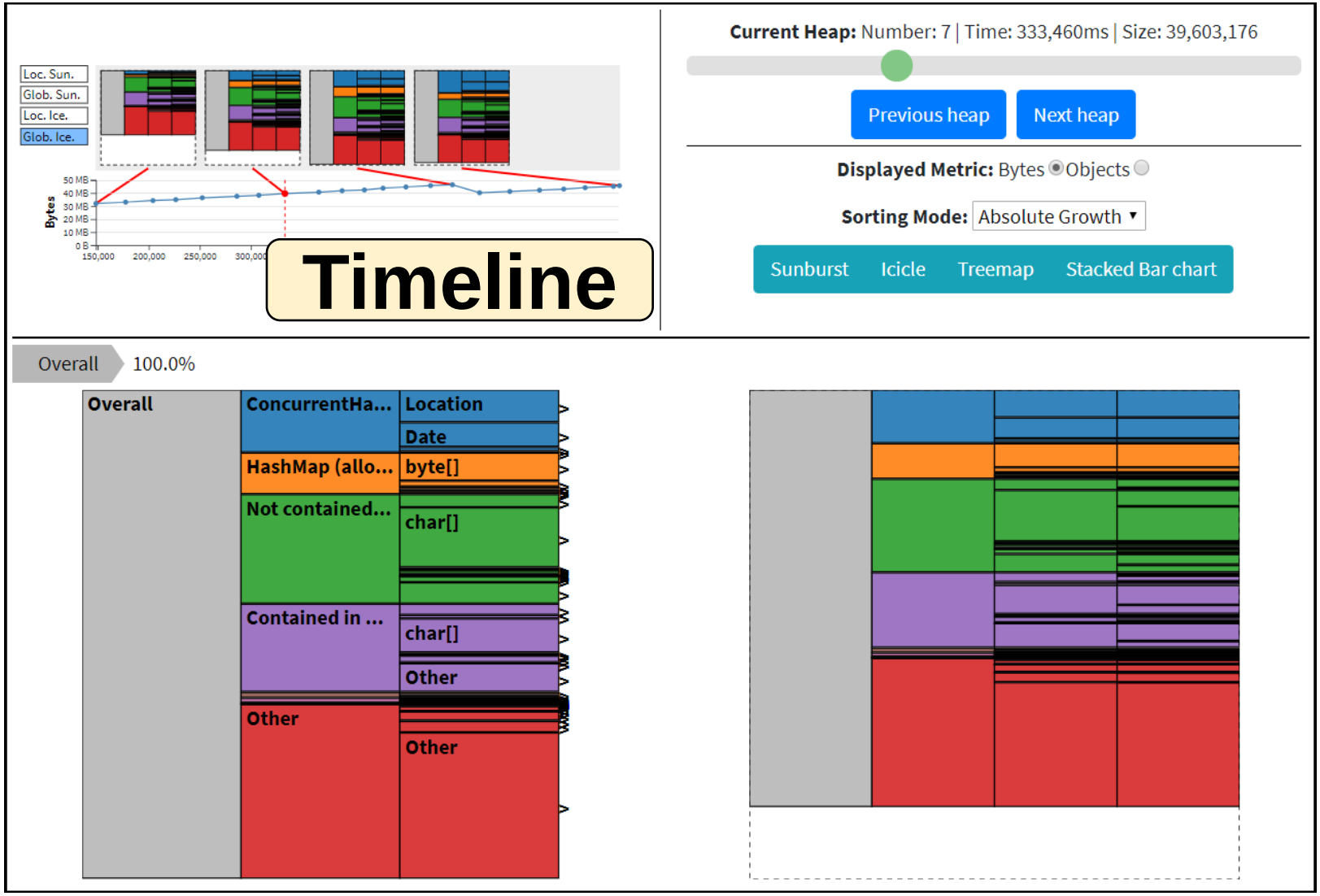


Small multiples at different points in time

PROTOTYPE AVAILABLE

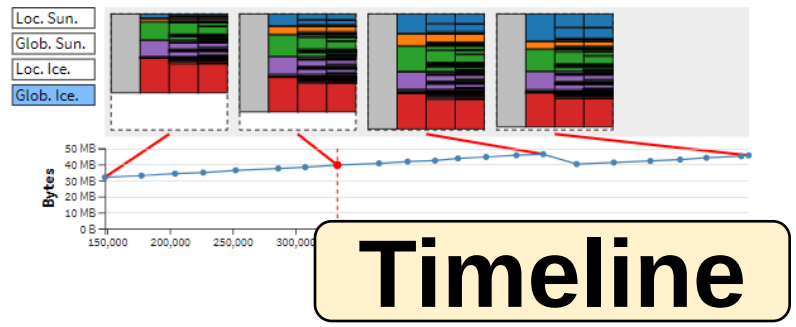


PROTOTYPE AVAILABLE

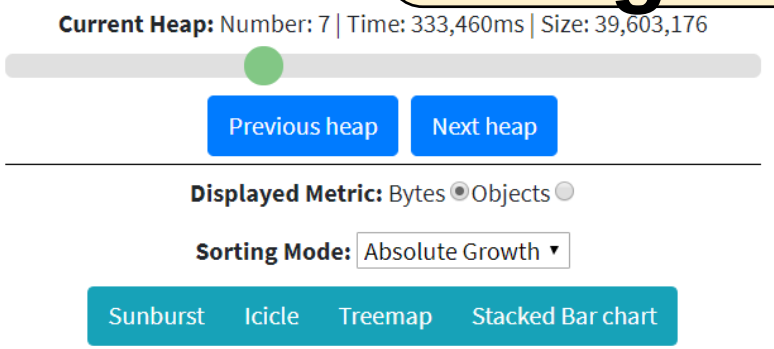


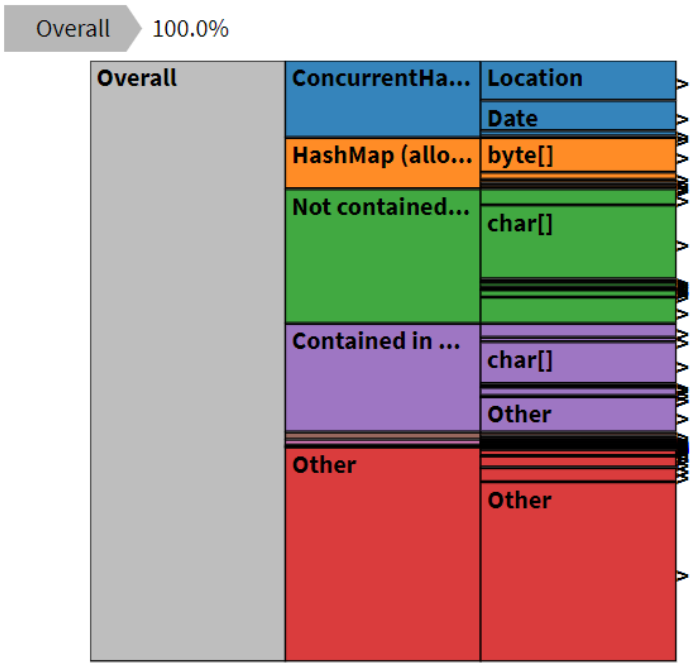
PROTOTYPE AVAILABLE

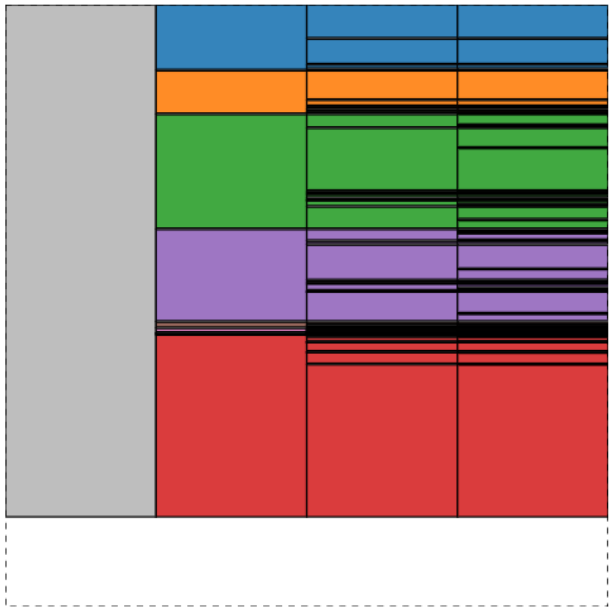
Time navigation



Timeline

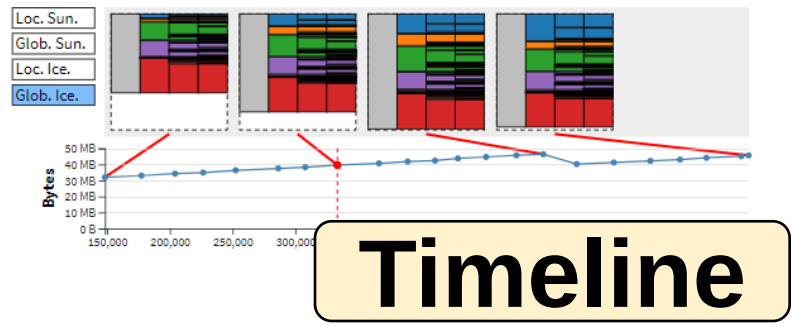




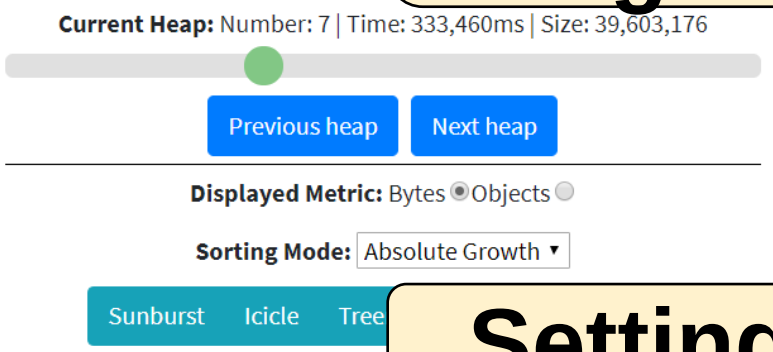


PROTOTYPE AVAILABLE

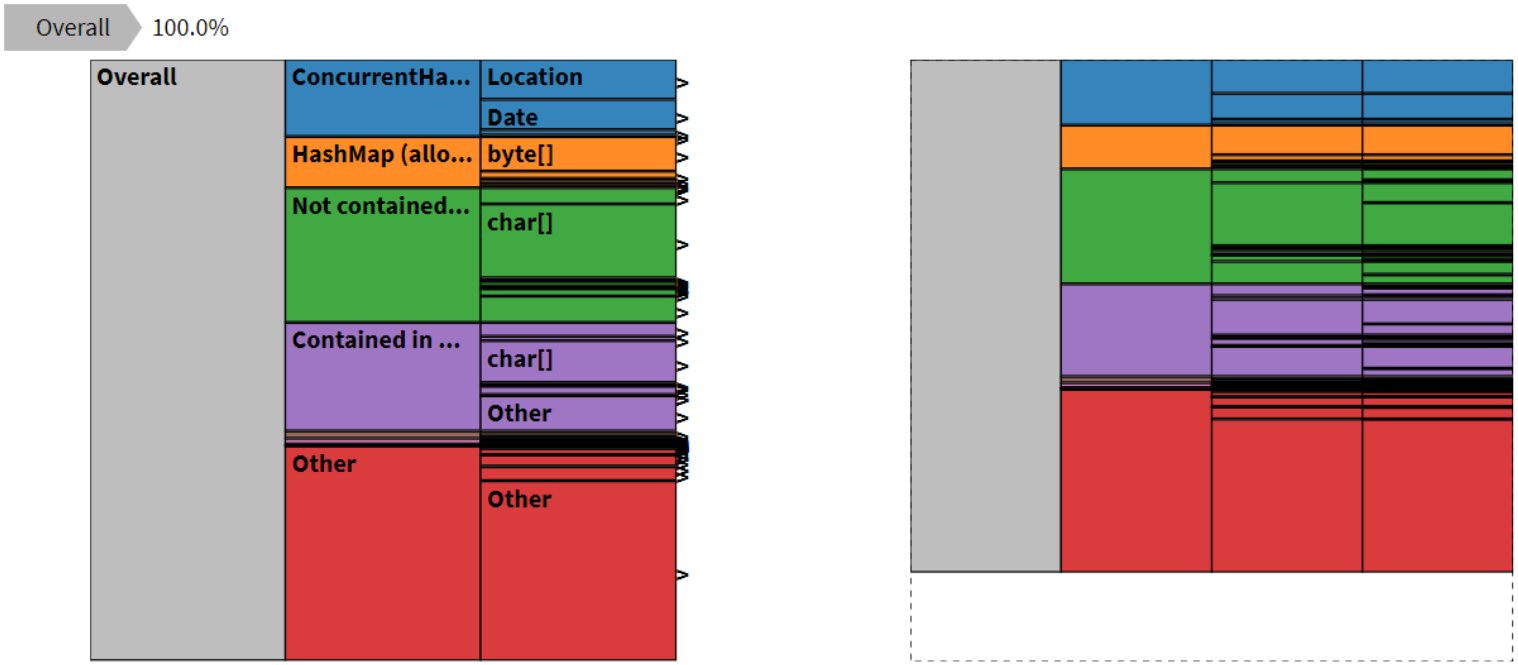
Time navigation



Timeline

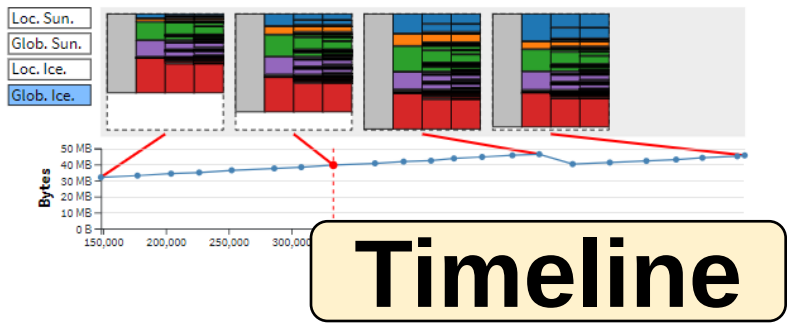


Settings

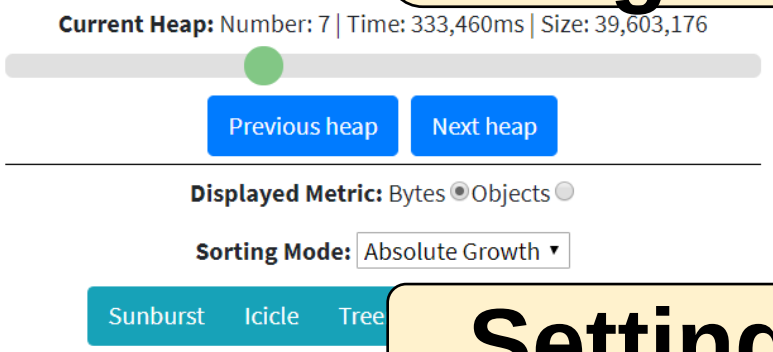


PROTOTYPE AVAILABLE

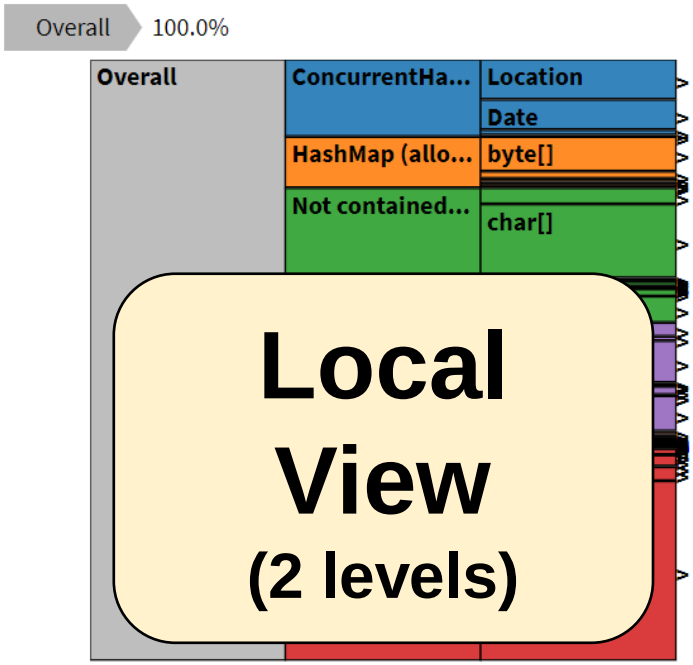
Time navigation



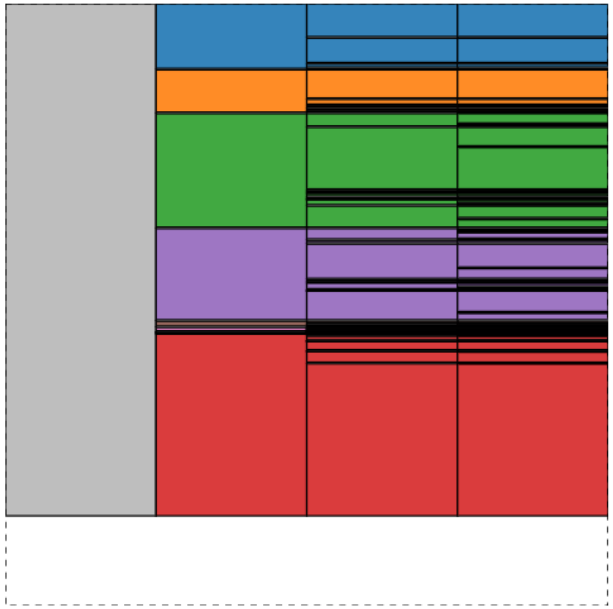
Timeline



Settings

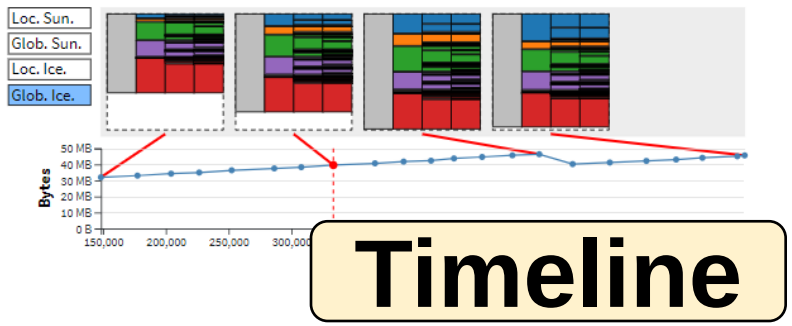


Local View (2 levels)

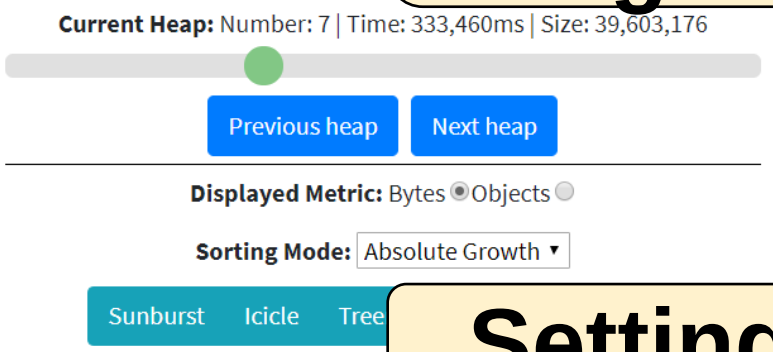


PROTOTYPE AVAILABLE

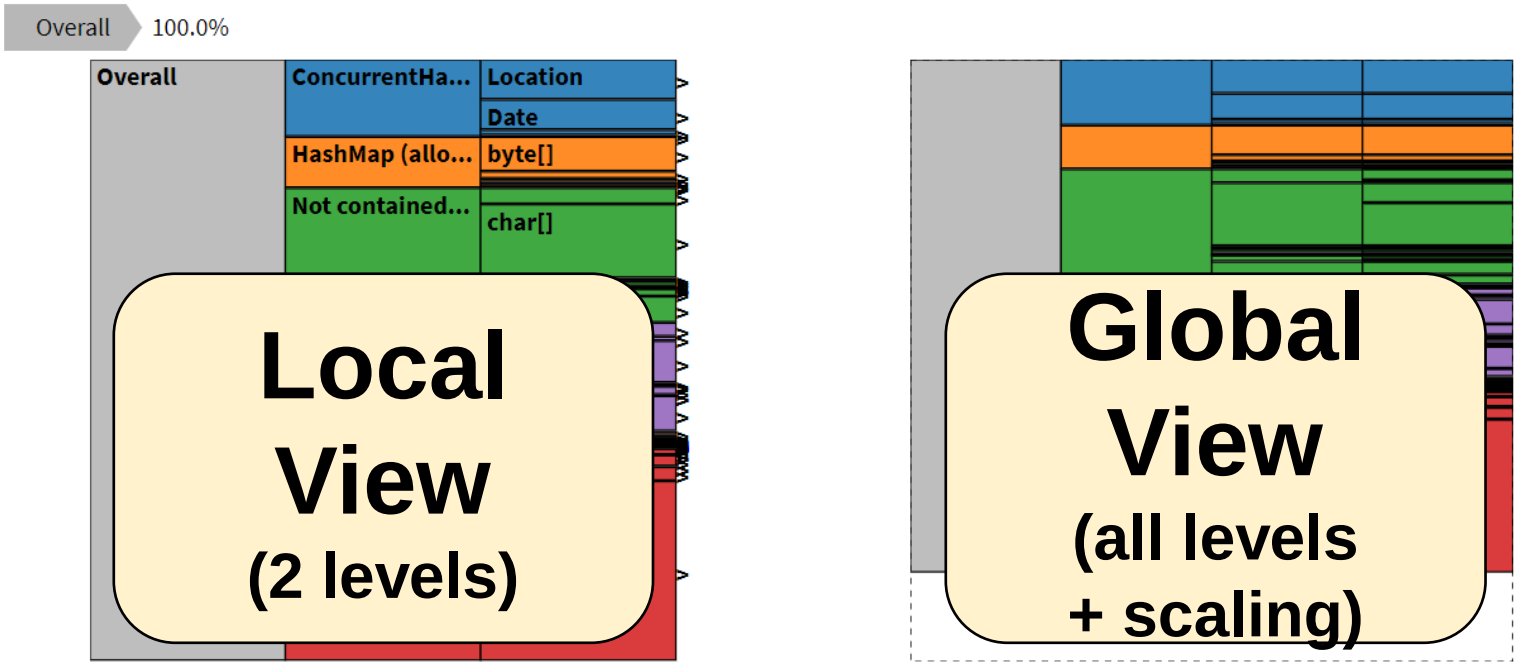
Time navigation



Timeline



Settings



Local View (2 levels)

Global View (all levels + scaling)

FUTURE WORK

Icons taken from https://fontawesome.com/ - license: https://fontawesome.com/license/free

FUTURE WORK

- Follow-up work accepted at STAG 2020
  „Memory Leak Analysis using Time-Travel-based and Timeline-based Tree Evolution Visualizations “

FUTURE WORK

- Follow-up work accepted at STAG 2020
  „Memory Leak Analysis using Time-Travel-based and Timeline-based Tree Evolution Visualizations “

STAG 2020
SMART TOOLS AND APPLICATIONS FOR GRAPHICS

- Guidance

Icons taken from https://fontawesome.com/ - license: https://fontawesome.com/license/free

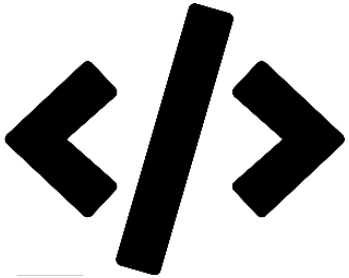
FUTURE WORK

- Follow-up work accepted at STAG 2020
  „Memory Leak Analysis using Time-Travel-based and Timeline-based Tree Evolution Visualizations “

STAG 2020
SMART TOOLS AND APPLICATIONS FOR GRAPHICS

- Guidance

- IDE Integration



Icons taken from https://fontawesome.com/ - license: https://fontawesome.com/license/free

TAKE-AWAYS

TAKE-AWAYS

Problem

How to visualize memory evolution (i.e., memory trees) over time?

TAKE-AWAYS

Problem	Tree Visualizations
How to visualize memory evolution (i.e., memory trees) over time?	Target audience: Novice memory analysts Sunburst Icicle

TAKE-AWAYS

Problem	Tree Visualizations	Evolution Over Time
How to visualize memory evolution (i.e., memory trees) over time?	Target audience: Novice memory analysts Sunburst Icicle	Stable layout „Relative position animation“ Don‘t destroy user‘s cognitive map

TAKE-AWAYS

Problem	Tree Visualizations	Evolution Over Time	Interaction
How to visualize memory evolution (i.e., memory trees) over time?	Target audience: Novice memory analysts Sunburst Icicle	Stable layout „Relative position animation“ Don‘t destroy user‘s cognitive map	Timeline view Time navigation Drill-down

TAKE-AWAYS

Problem	Tree Visualizations	Evolution Over Time	Interaction
How to visualize memory evolution (i.e., memory trees) over time?	Target audience: Novice memory analysts Sunburst Icicle	Stable layout „Relative position animation“ Don‘t destroy user‘s cognitive map	Timeline view Time navigation Drill-down

Prototype available at
https://bit.ly/
STAG-MemoryTreeVizTool

Video available at
https://bit.ly/
STAG-MemoryTreeVizVideo

TAKE-AWAYS

Problem	Tree Visualizations	Evolution Over Time	Interaction
How to visualize memory evolution (i.e., memory trees) over time?	Target audience: Novice memory analysts Sunburst Icicle	Stable layout „Relative position animation“ Don‘t destroy user‘s cognitive map	Timeline view Time navigation Drill-down

Markus Weninger

Johannes Kepler University
Linz, Austria

markus.weninger@jku.at

Prototype available at
https://bit.ly/
STAG-MemoryTreeVizTool

Video available at
https://bit.ly/
STAG-MemoryTreeVizVideo