



Dynamic Shape and Data Structure Analysis in Java

Presented by Sokhom Pheng
(Supervised by Clark Verbrugge)

McGill University
October 17th 2005



Outline

- Introduction & Contribution
- Design
- Challenges
- Experimental Results
- Future work & Conclusion



Outline

- Introduction & Contribution
- Design
- Challenges
- Experimental Results
- Future work & Conclusion



Introduction

Have you ever wondered how data structures were constructed in your programs?

- Two approaches
 - Static analysis
 - Conservative
 - Limited to simple situations
 - Dynamic analysis
 - More accurate
 - More expensive



Contribution

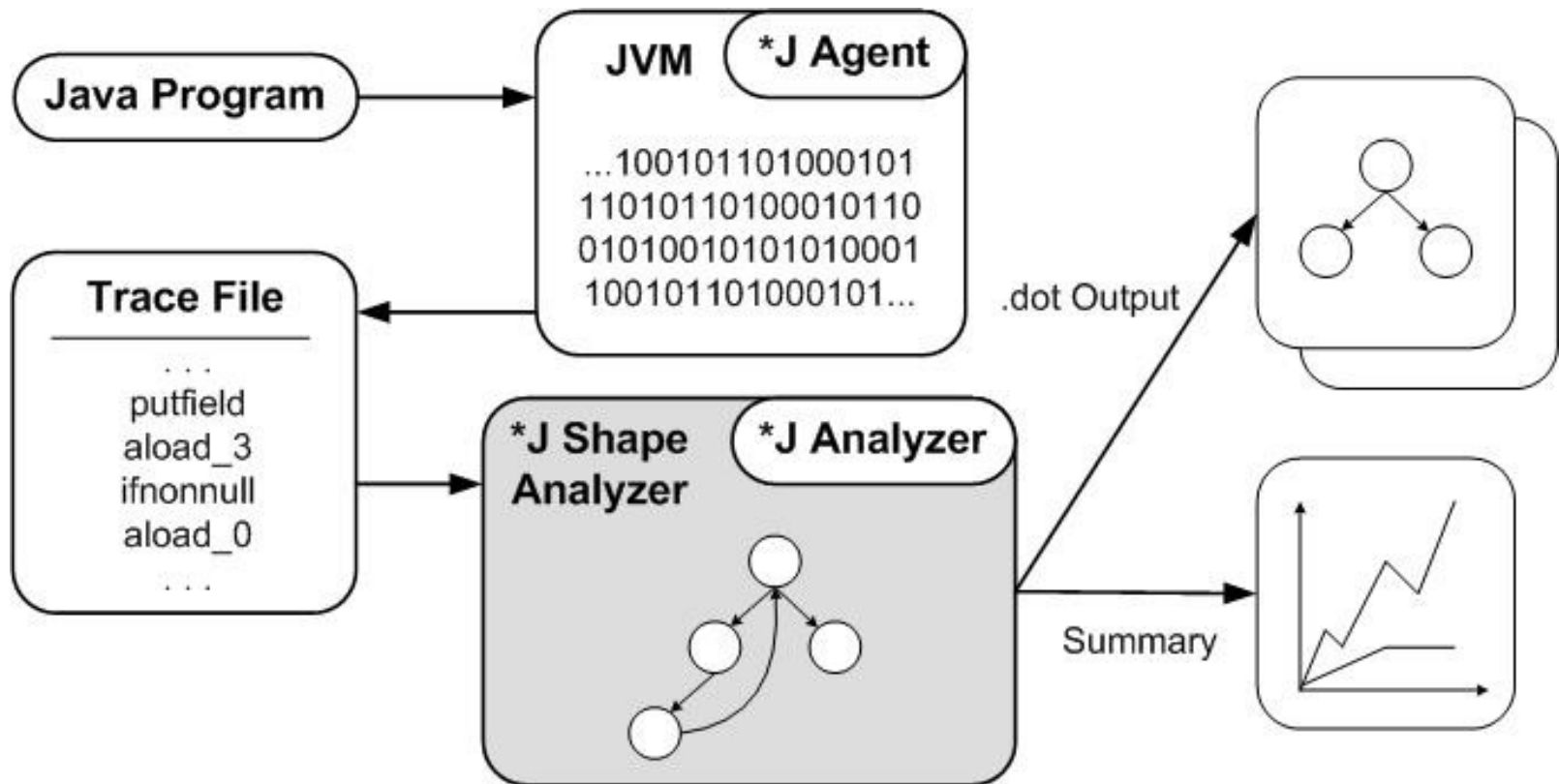
- Framework for data structure visualization
 - Program understanding
 - Set limits on static approaches
- Provide two techniques for visualization
- Provide dynamic representation of garbage data



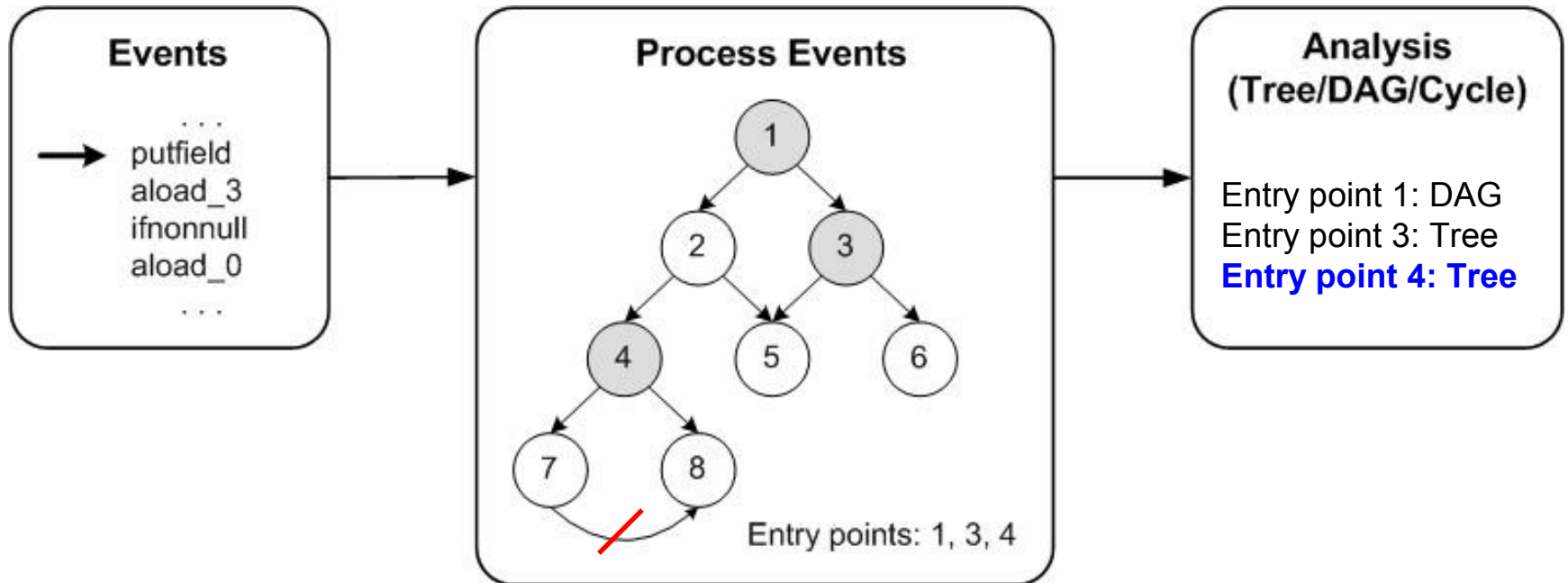
Outline

- Introduction & Contribution
- **Design**
- Challenges
- Experimental Results
- Future work & Conclusion

Design Overview



*J Shape Analyzer

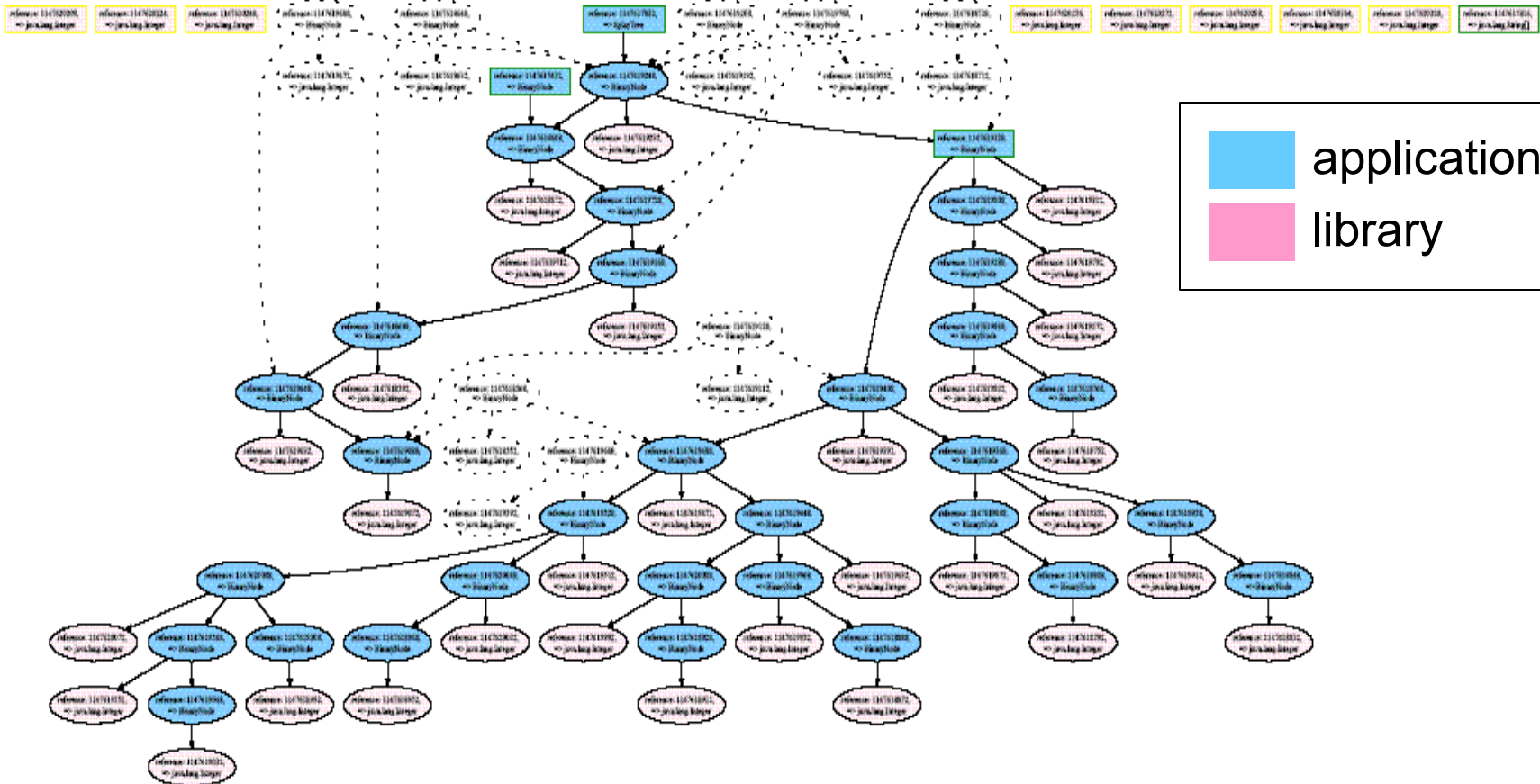




More Analysis Info

- GC info
 - Might be useful for GC optimization
 - If visualized, can see drag effect
- Aging
 - Useful for general program understanding
 - Useful for teaching

GC Info (cont'd)



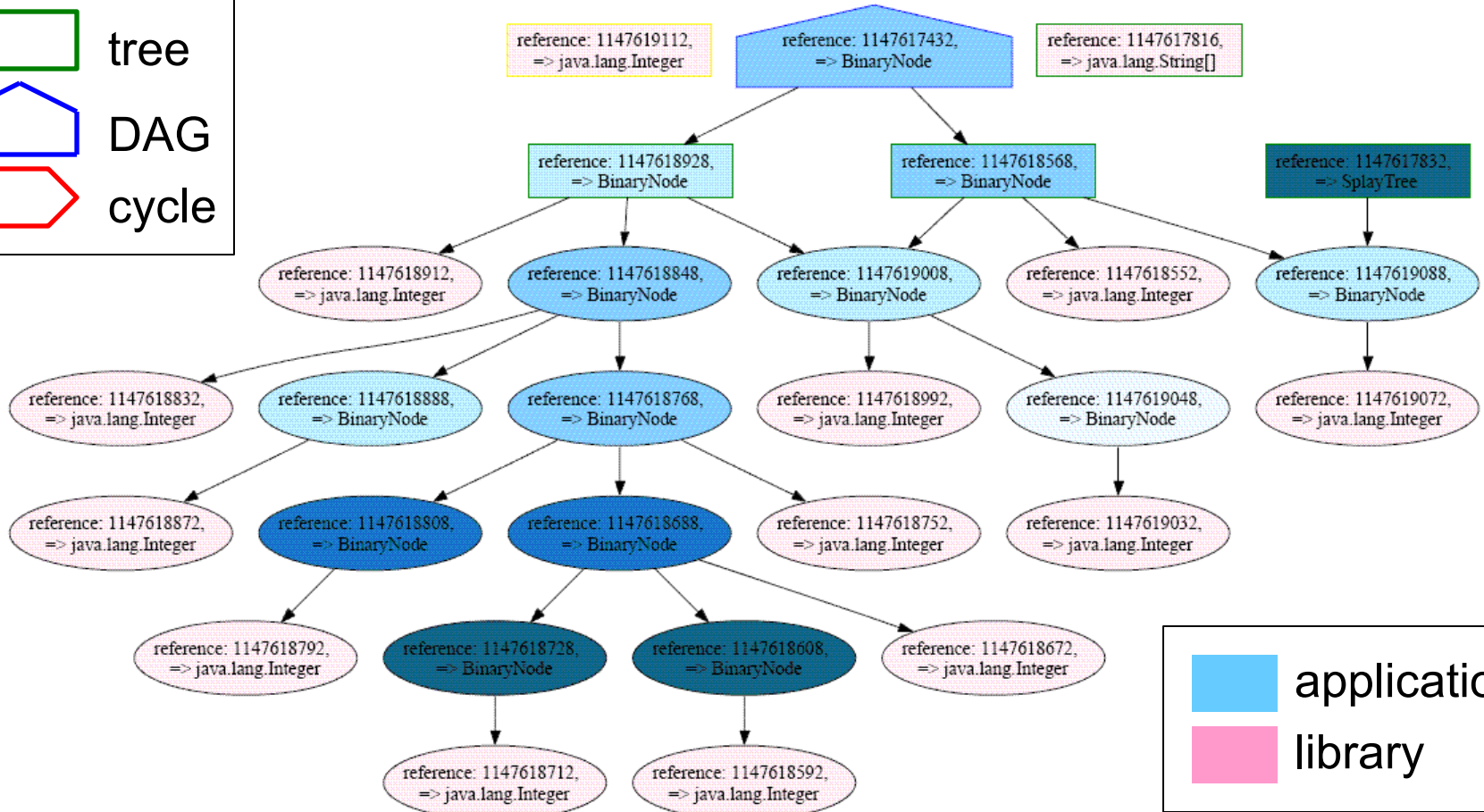
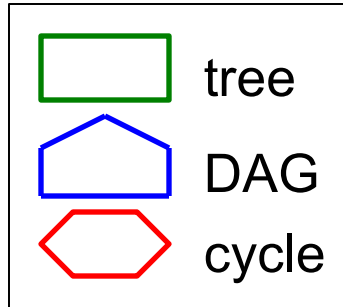
GC Info (cont'd)

reference: 1147620240,
=> java.lang.Integer

reference: 1147619688,
=> BinaryNode

reference: 1147619672,
=> java.lang.Integer

Aging (splay tree)



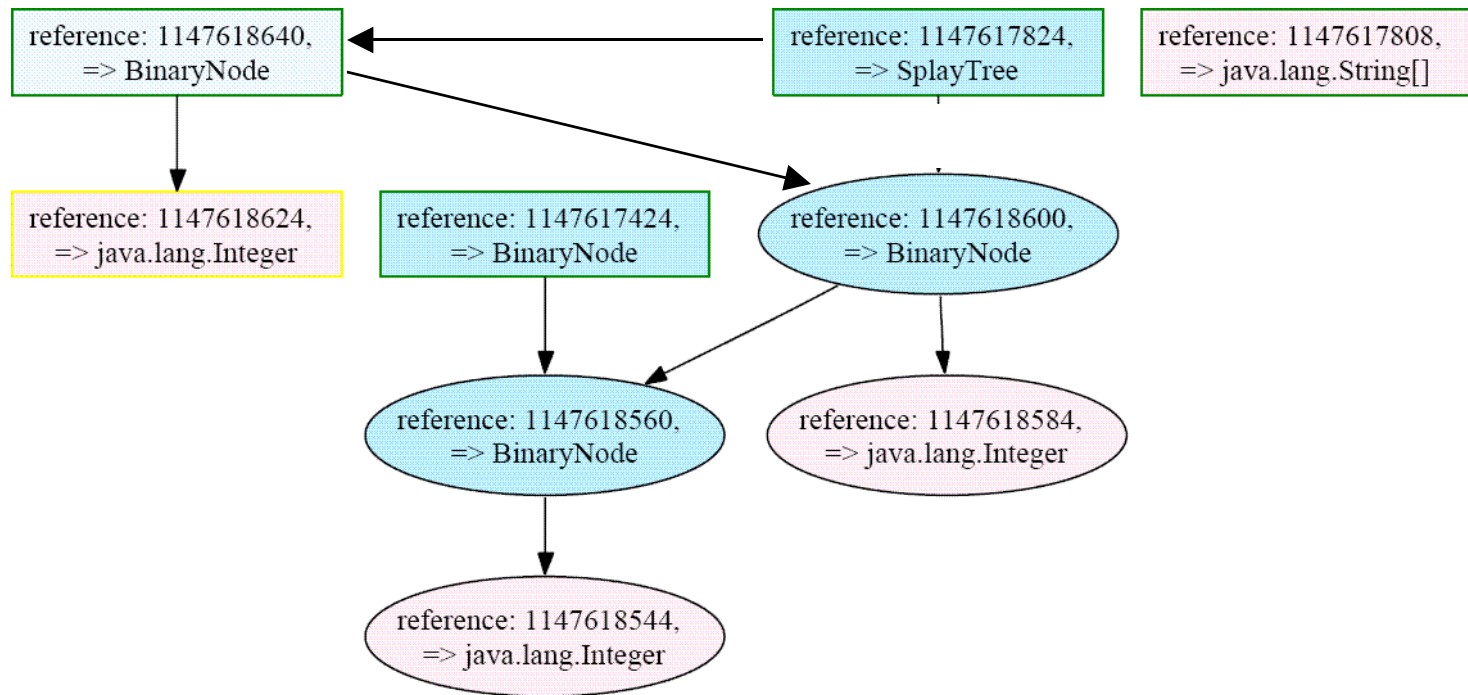


Outline

- Introduction & Contribution
- Design
- **Challenges**
- Experimental Results
- Future work & Conclusion

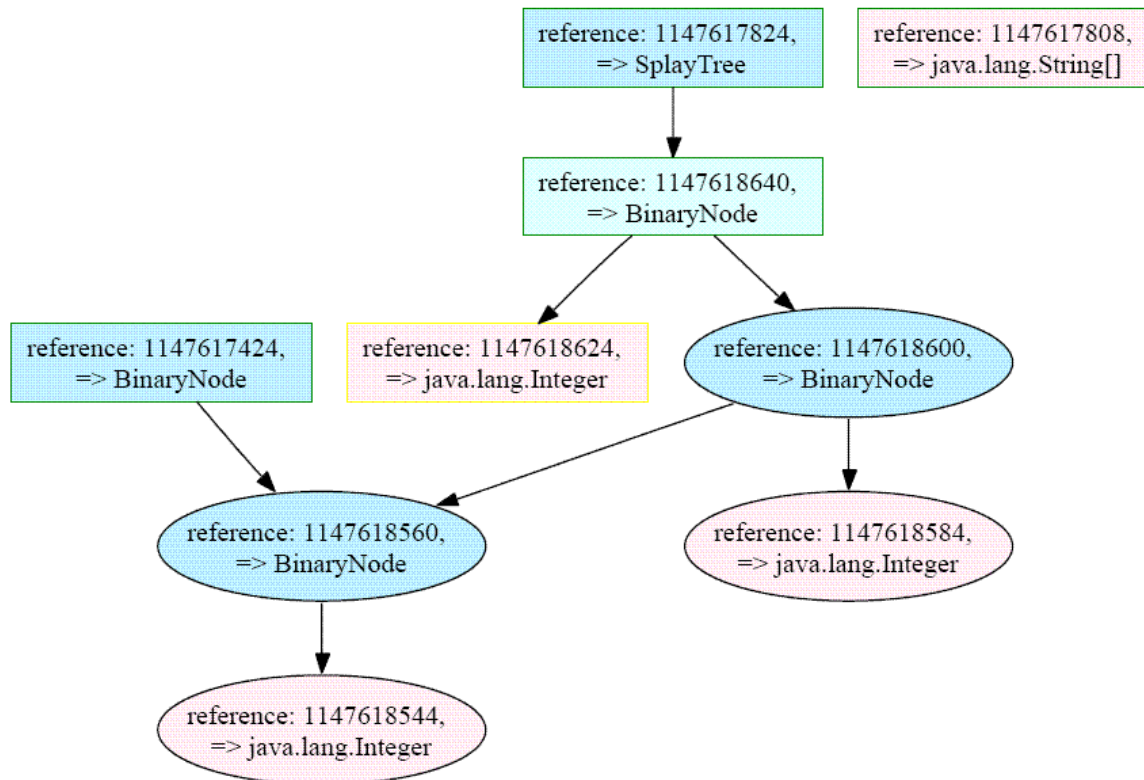
Challenge 1: Animation Issue

This is what we want:



Animation Issue (cont'd)

This is what we have:

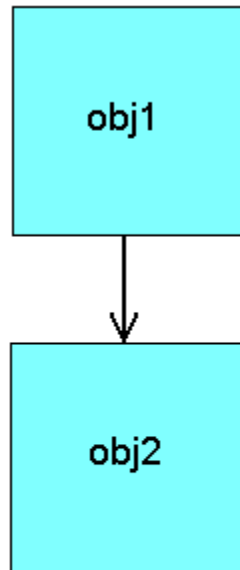




Tools Used

- Tom Sawyer & yWorks yFiles
 - Did not work properly as incremental
- Graphviz Neato
 - Pin down object causes problems
- Graphviz Dot
 - Tool we use
 - Nicest and easiest to use

Tom Sawyer & yWorks yFiles



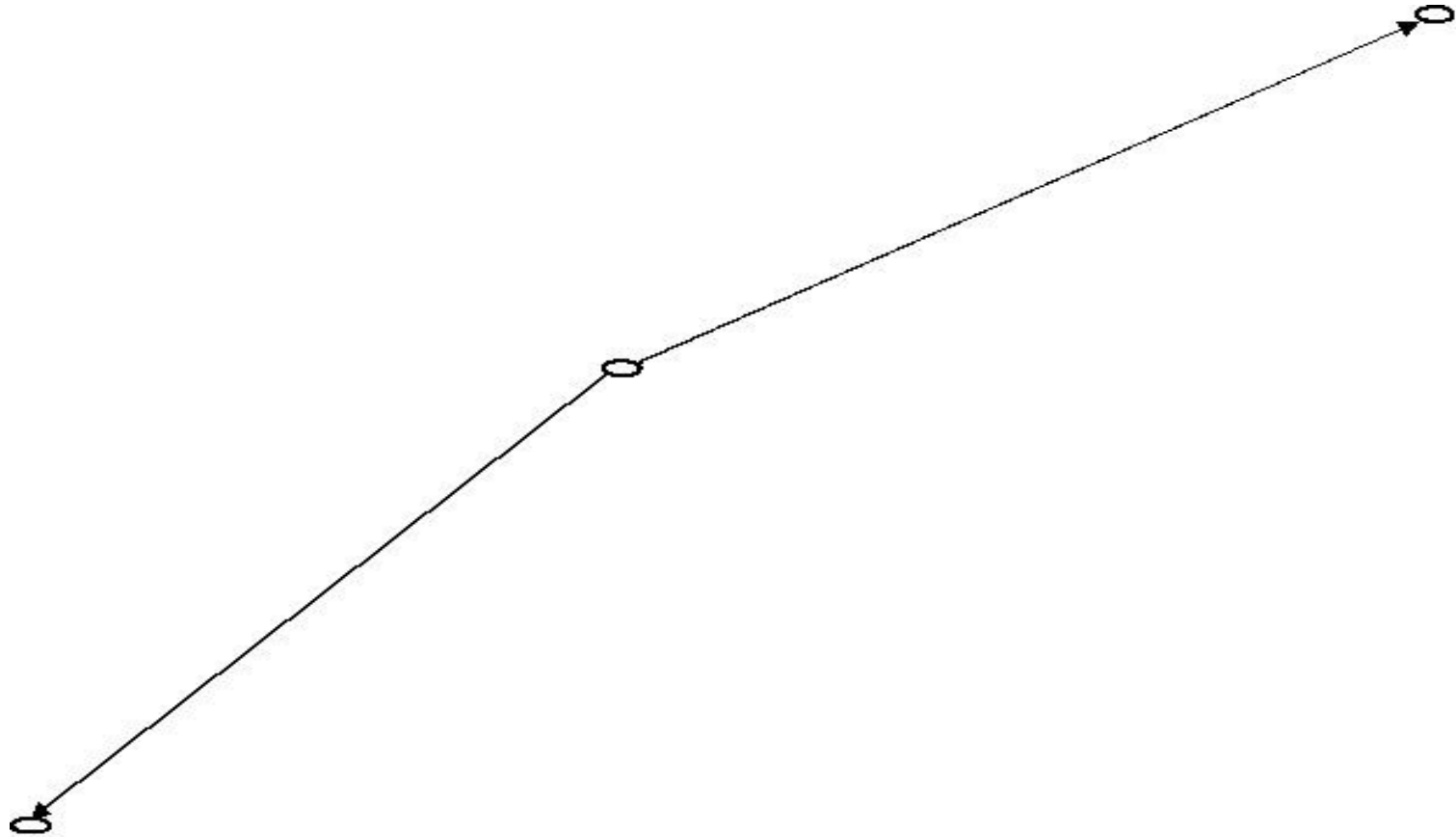


Tools Used

- Tom Sawyer & yWorks yFiles
 - Did not work properly as incremental
- **Graphviz Neato**
 - Pin down object causes problems
- Graphviz Dot
 - Tool we use
 - Nicest and easiest to use



Graphviz Neato





Tools Used

- Tom Sawyer & yWorks yFiles
 - Did not work properly as incremental
- Graphviz Neato
 - Pin down object causes problems
- **Graphviz Dot**
 - Tool we use
 - Nicest and easiest to use



Challenge 2: Data Size

- Visualization limitation
 - Too many objects to draw
 - BiSort: > 120,000 objects
 - Too many snapshots generated
 - Jess: > 48,000,000 snapshots
- Computational cost
 - Analysis



Data Size Potential Solutions

- Problem with number of snapshots and cost
 - Analyze data structure after a certain number of modifications instead
- Problem with number of objects
 - Numerical summary



Outline

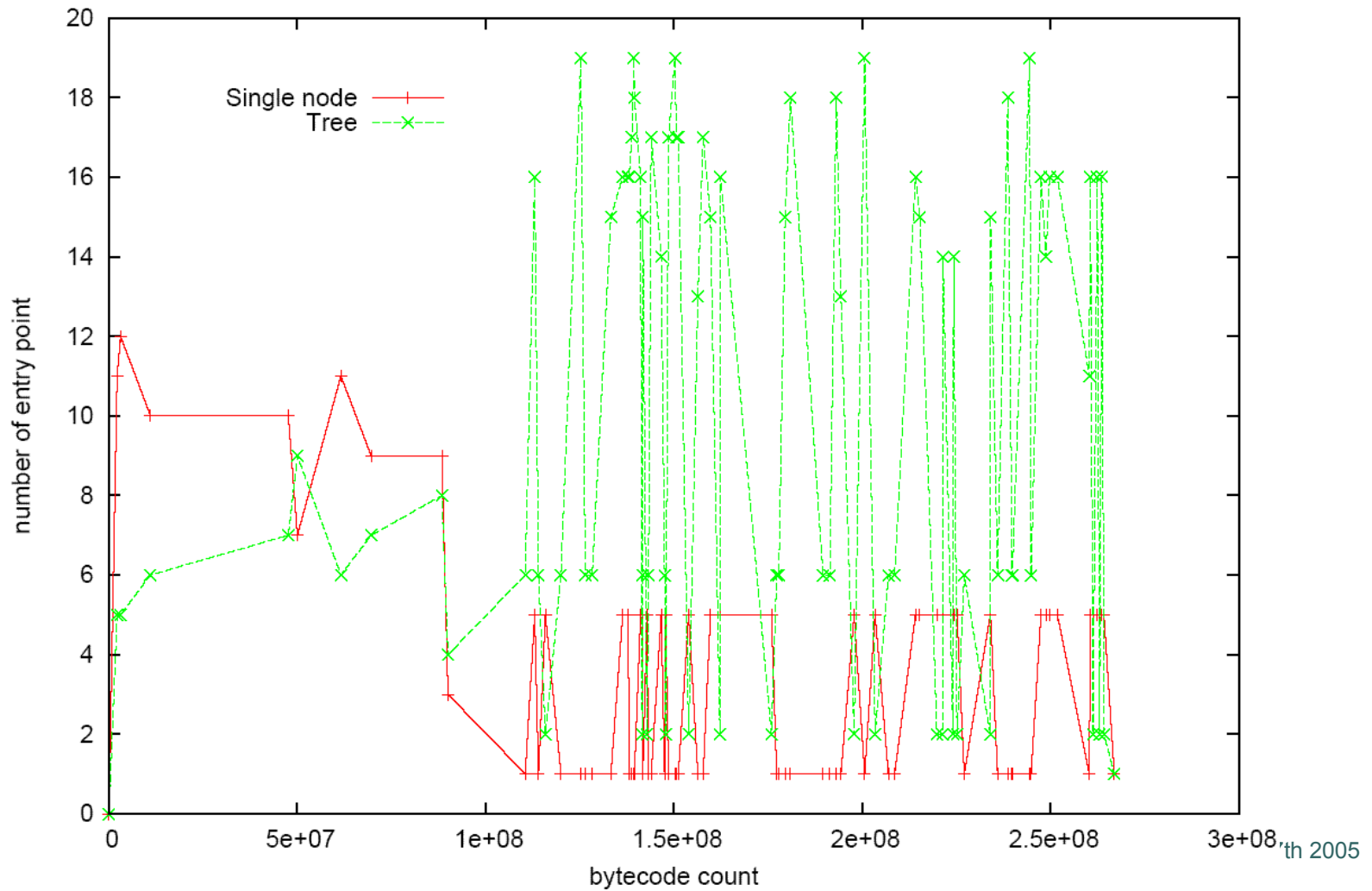
- Introduction & Contribution
- Design
- Challenges
- **Experimental Results**
- Future work & Conclusion



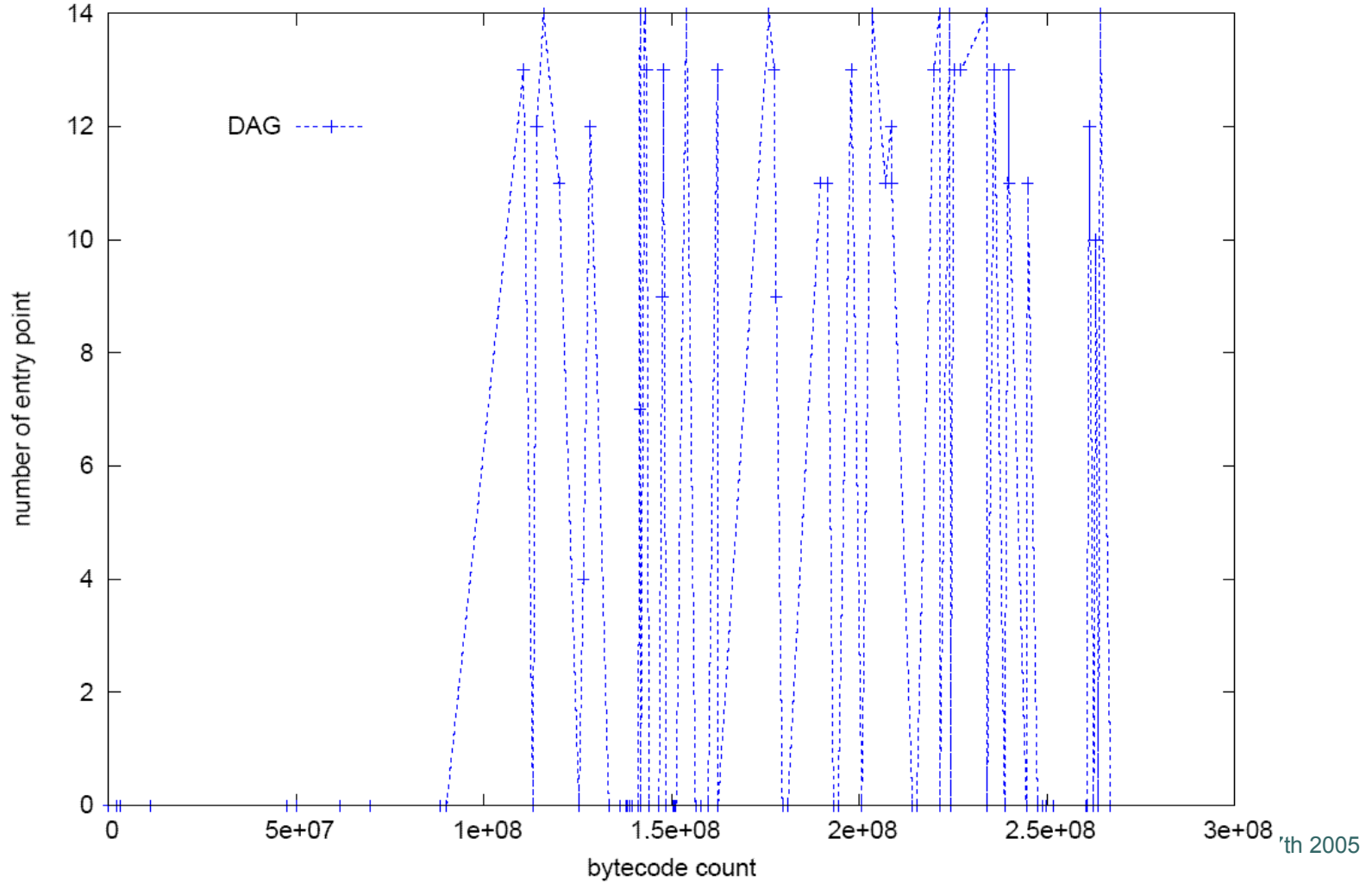
Benchmarks

- Tiny
 - Splay tree, red-black tree
- Small
 - JOlden suite
- Large
 - SPECjvm98 suite

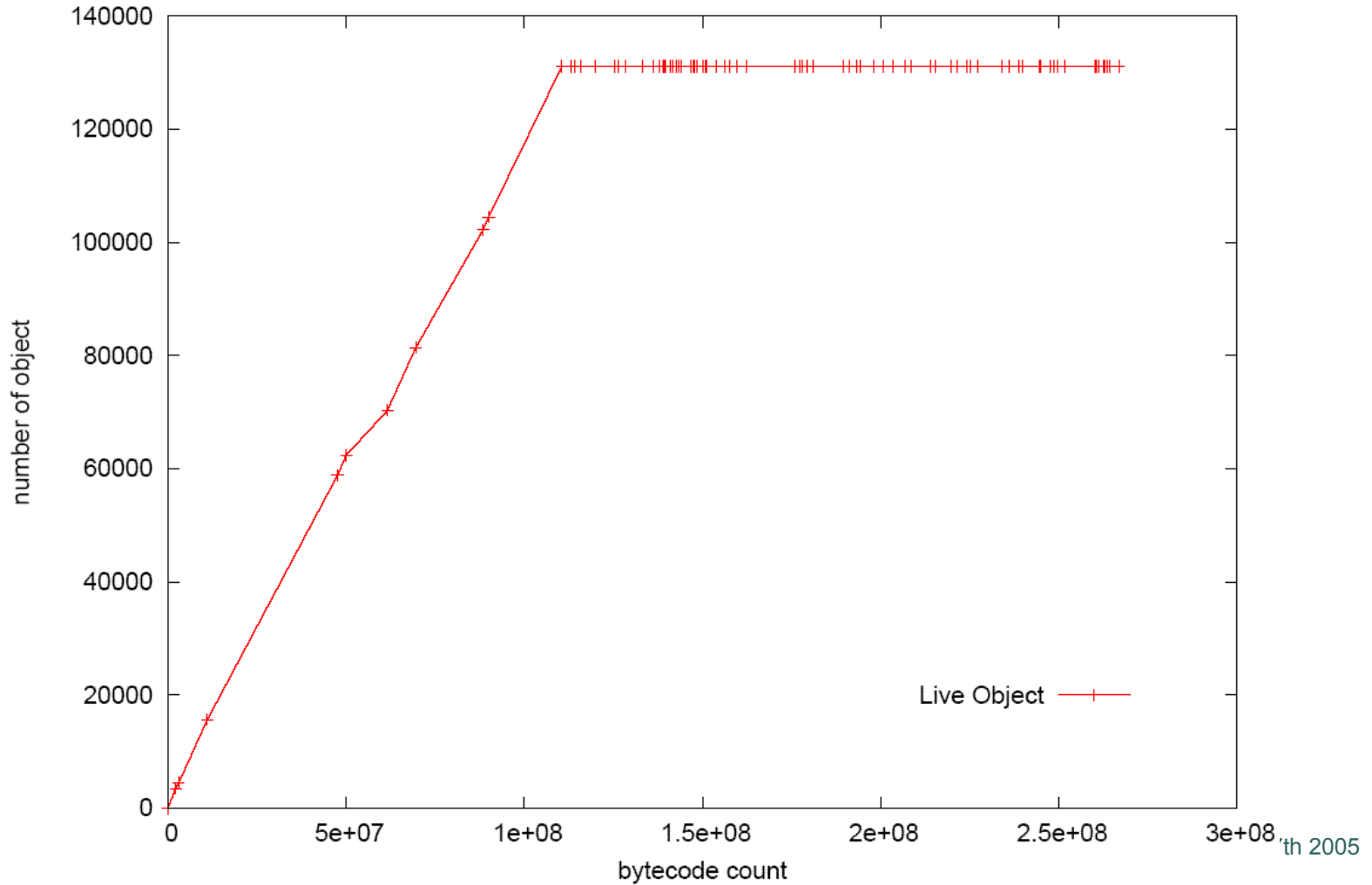
JOlden - BiSort



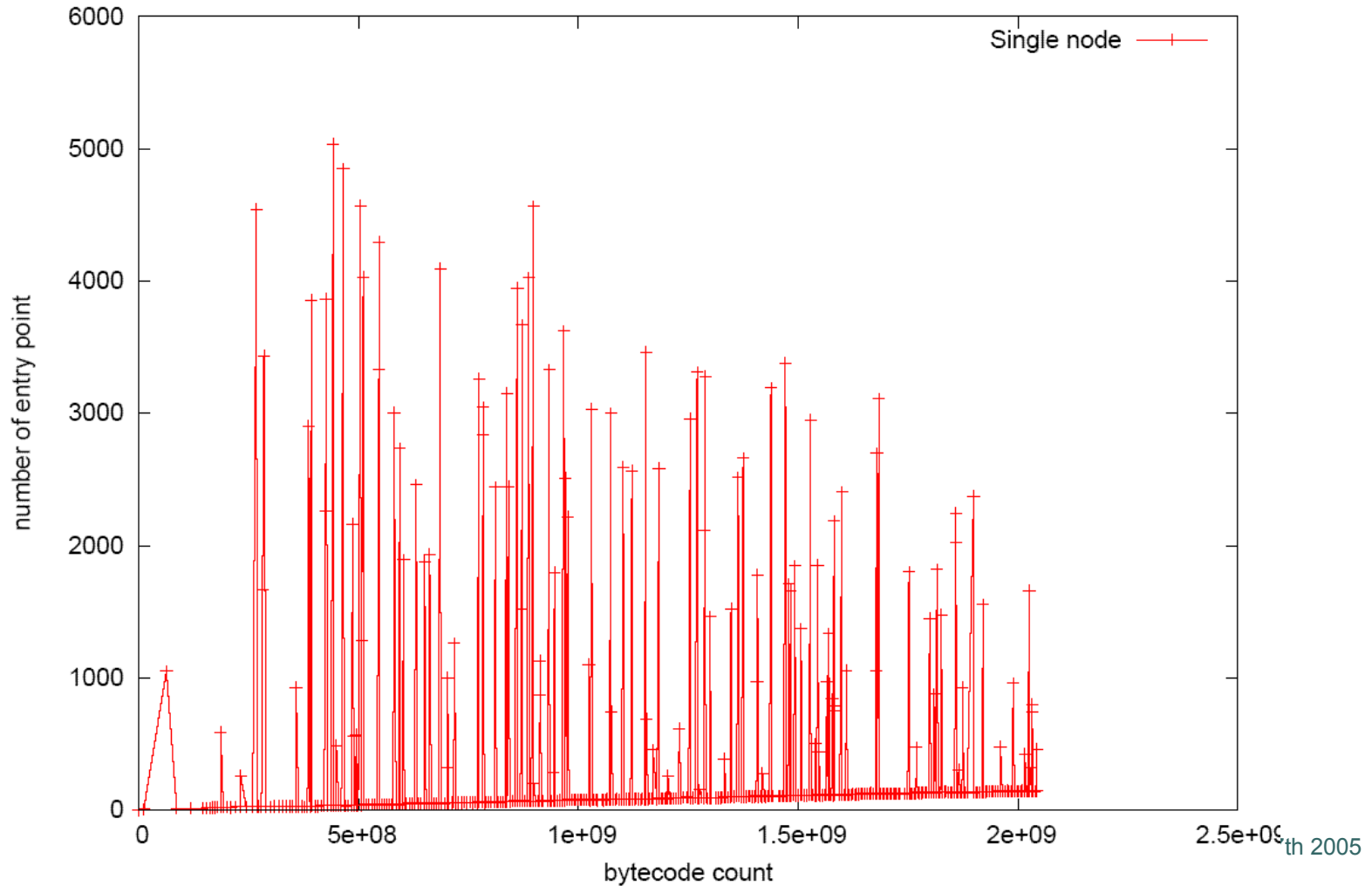
BiSort (cont'd)



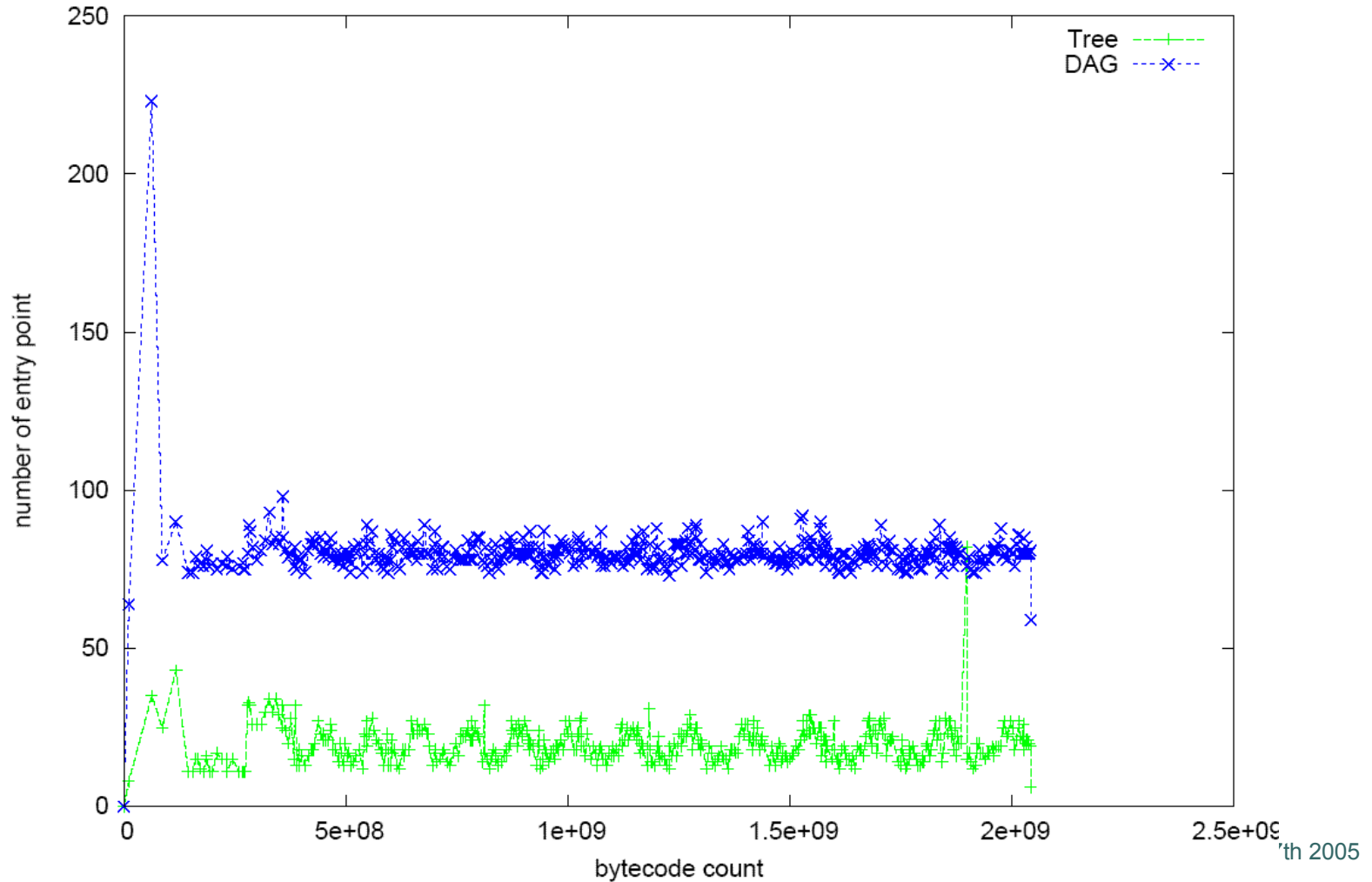
BiSort GC Info



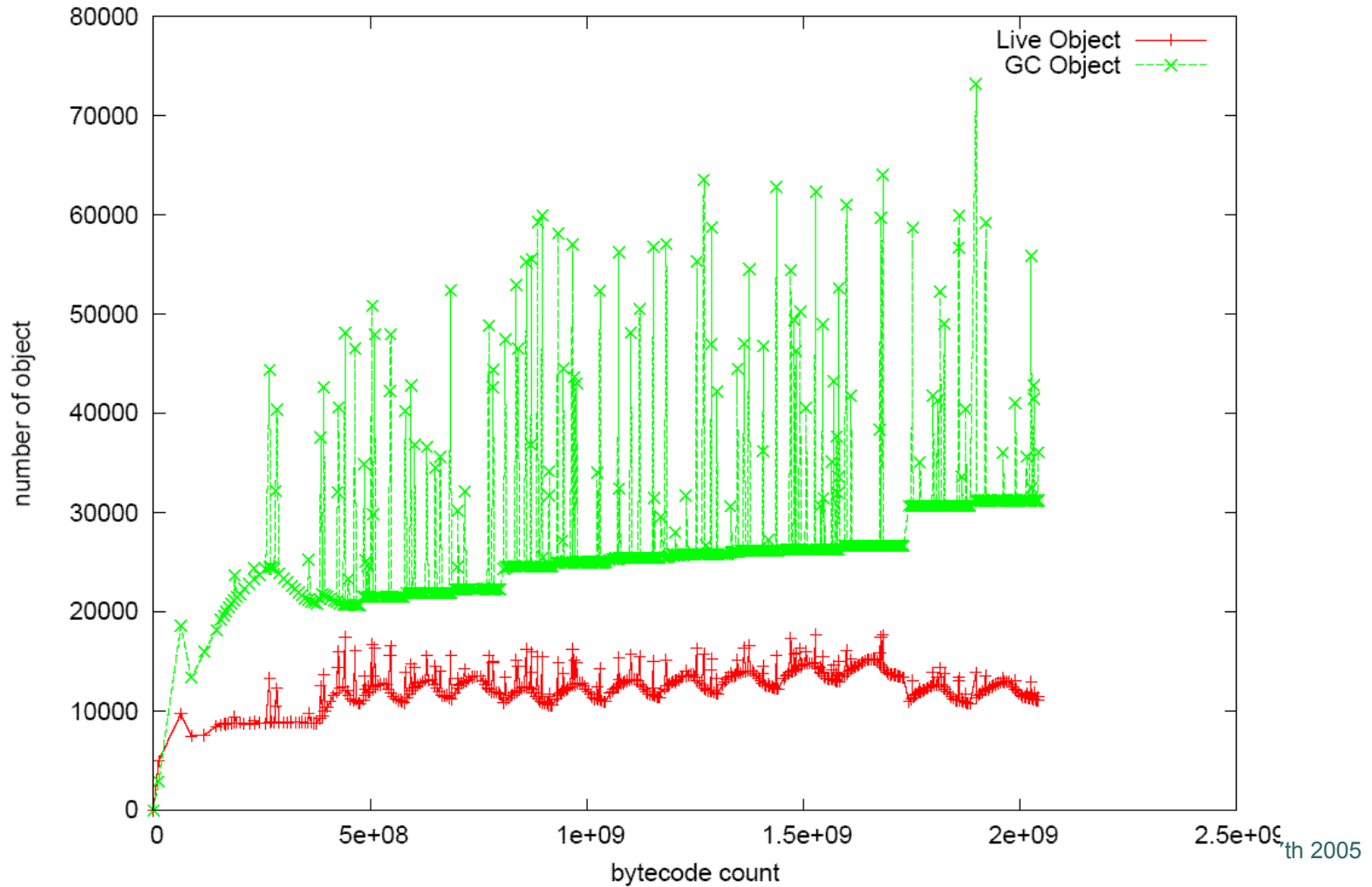
SPECjvm98 - Jess



Jess (cont'd)



Jess GC info





Outline

- Introduction & Contribution
- Design
- Challenges
- Experimental Results
- **Future work & Conclusion**



Future Work

- Map to static code location
- Visual improvement
 - Improve animation quality
 - Collapsing nodes to look at bigger graphs
- More analyses, i.e. summary graphs



Related Work

○ Shape Analysis

● With Annotation

- Hummel et al.: program annotation
- Fradet et Le Métayer: language annotation

● Without annotation

- Ghiya and Hendren: tree/DAG/cycle
- Wilhelm et al.: shape graph
- Navarro et al.: reference-shape graph



Related Work (cont'd)

○ Dynamic Analysis

● Online

- MIT Program Analysis Group:
The Daikon invariant detector
- The Dynamo Project (Indiana University)

● Offline

- Dufour: *J, a tool for dynamic analysis of Java programs



Conclusion

- Design
- Challenges
 - Visualization & tools
 - Amount of data
- Experimental Results
 - JOlden
 - SPECjvm98



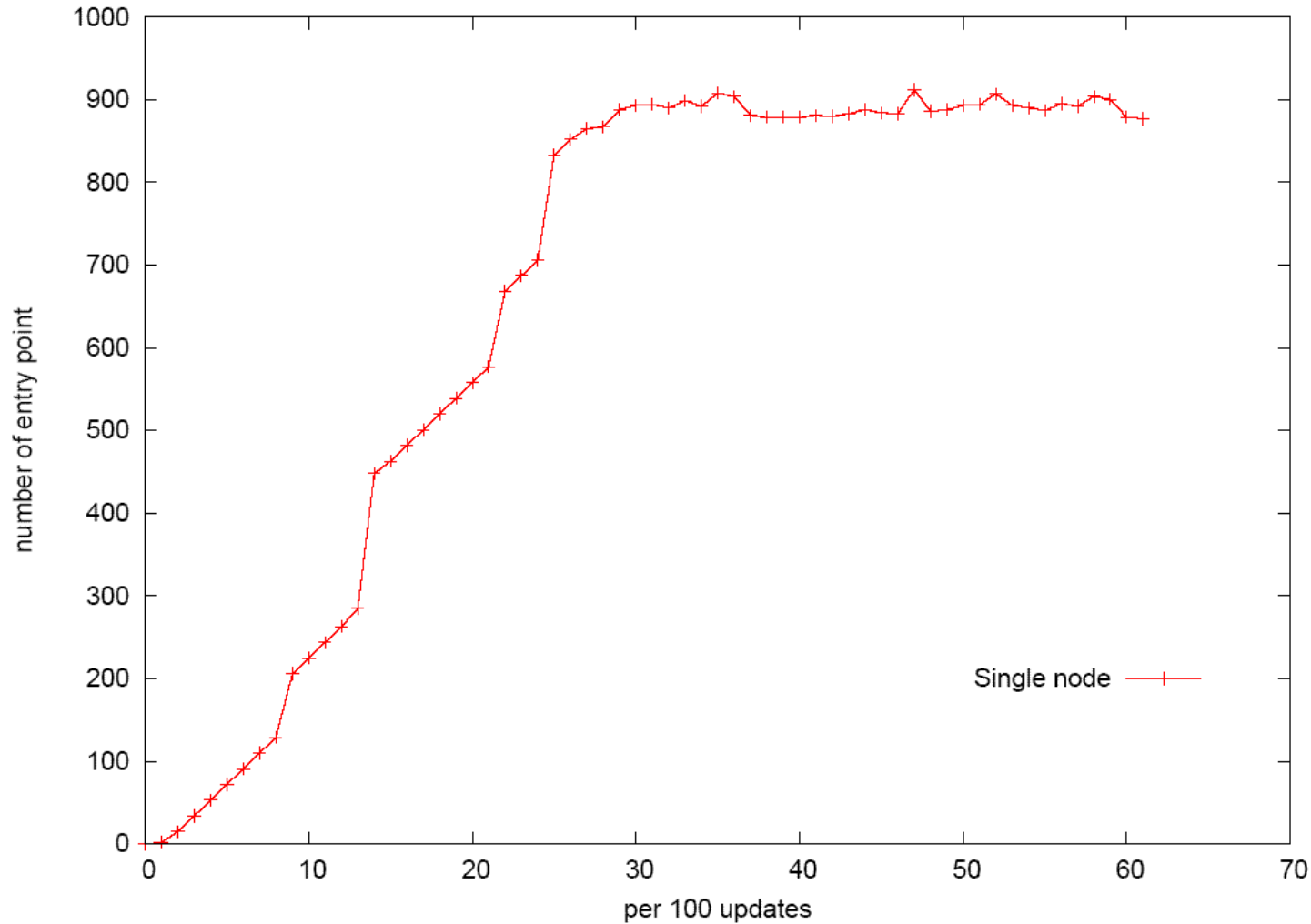
References

1. J. Bogda and A. Singh. Can Shape Analysis work at run-time? In *Proceedings of the 1st Java Virtual Machine Research and Technology Symposium, April 23-24, 2001*.
2. B. Dufour, K. Driesen, L. Hendren, and C. Verbrugge. Dynamic metrics for java. OOPSLA '03, pages 149-168.
3. P. Fradet and D. L. Métayer. Shape types. POPL '97.
4. R. Ghiya and L. Hendren. Is it a tree, a dag, or a cyclic graph? A shape analysis for heap-directed pointers in c. POPL '96.
5. B. Hackett and R. Rugina. Region-based shape analysis with tracked locations. POPL '05.
6. A. Navarro, F. Corbera, R. Asenjo, A. Tineo, O. Plata, and E. Zapata. A new dependence test based on shape analysis for pointer-based codes. LCPC '04
7. R. Wilhelm, S. Sagiv, and T. W. Reps. Shape analysis. In *Computational Complexity*, pages 1-17, 2000.

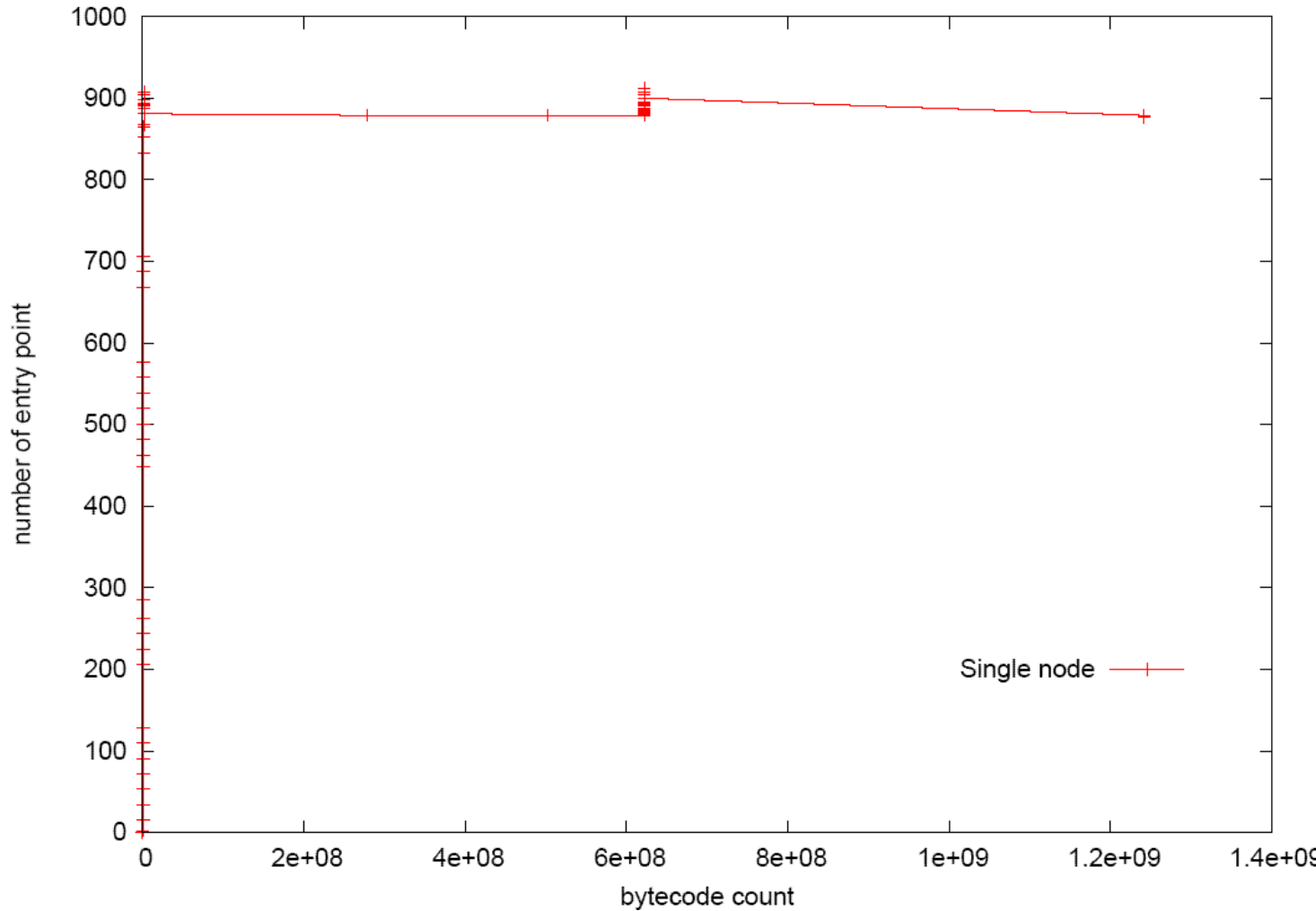


Questions?

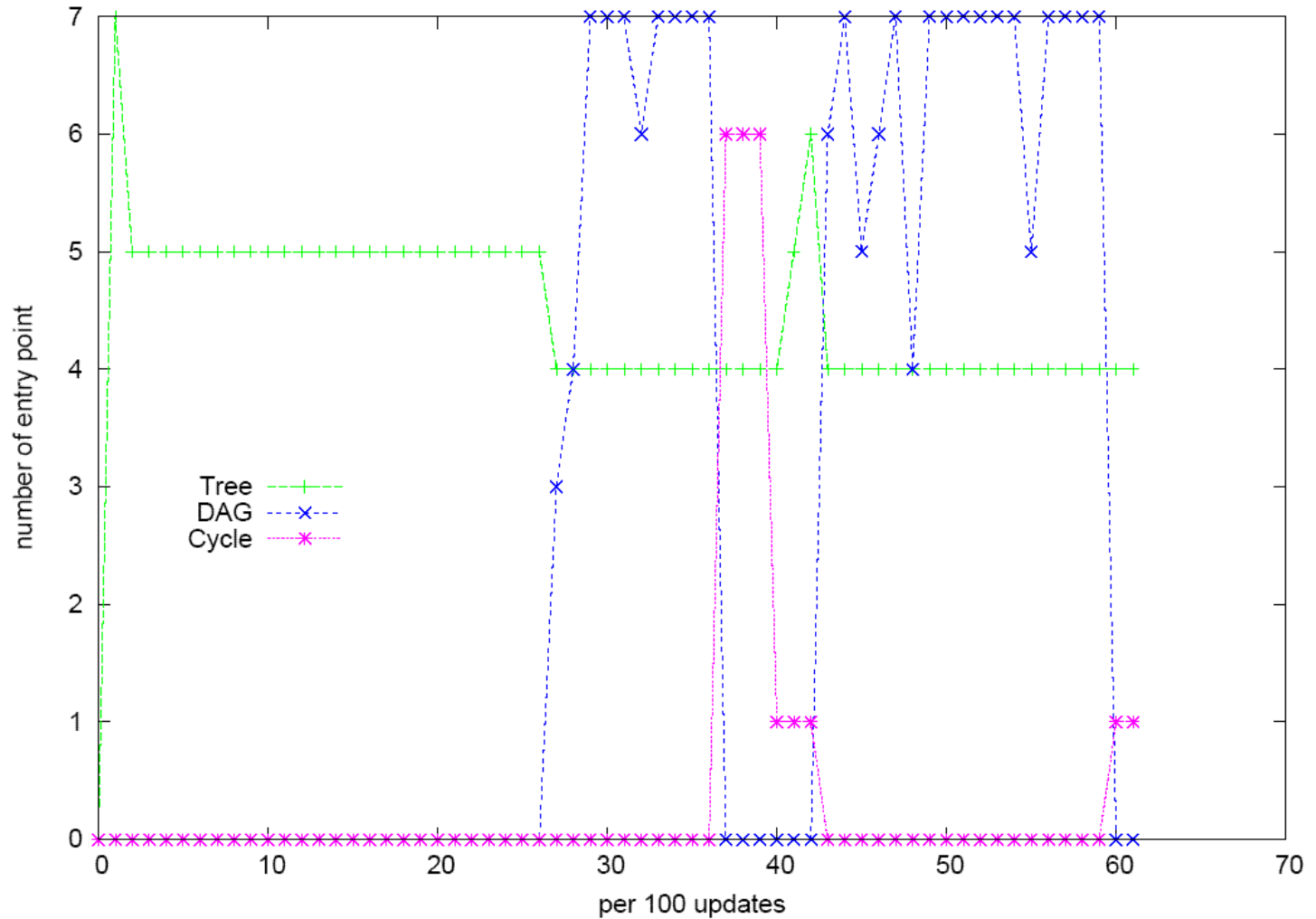
SPECjvm98 - MpegAudio



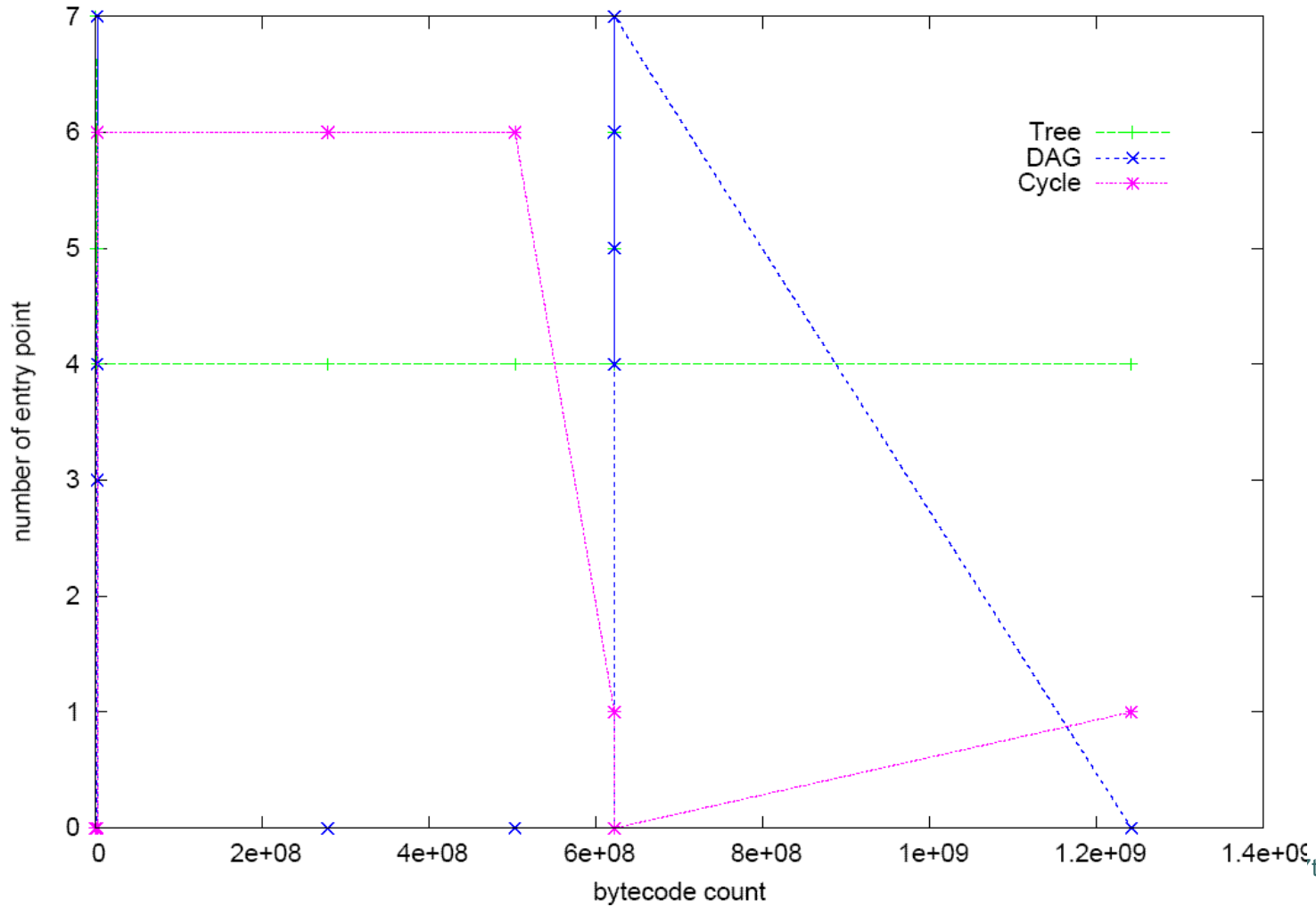
MpegAudio (cont'd)



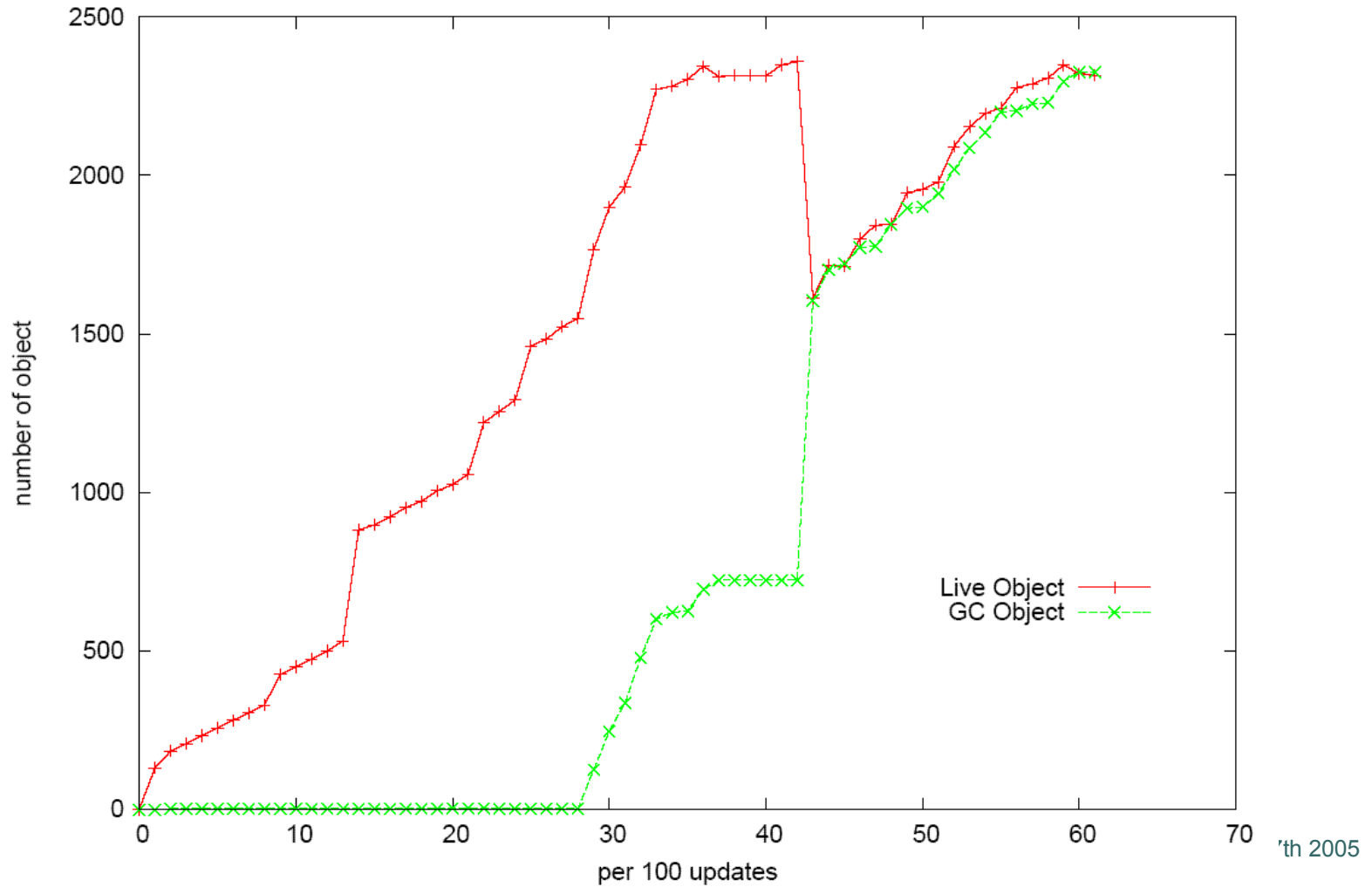
MpegAudio (cont'd)



MpegAudio (cont'd)



MpegAudio GC Info



MpegAudio GC Info (cont'd)

