Evolution of the partial history tree

Abdellah Rahmani

May 16th, 2022

Polytechnique Montreal

DORSAL Laboratory

Agenda

- The partial history tree
- Contribution and work in progress
- What’s next ?
- Conclusion

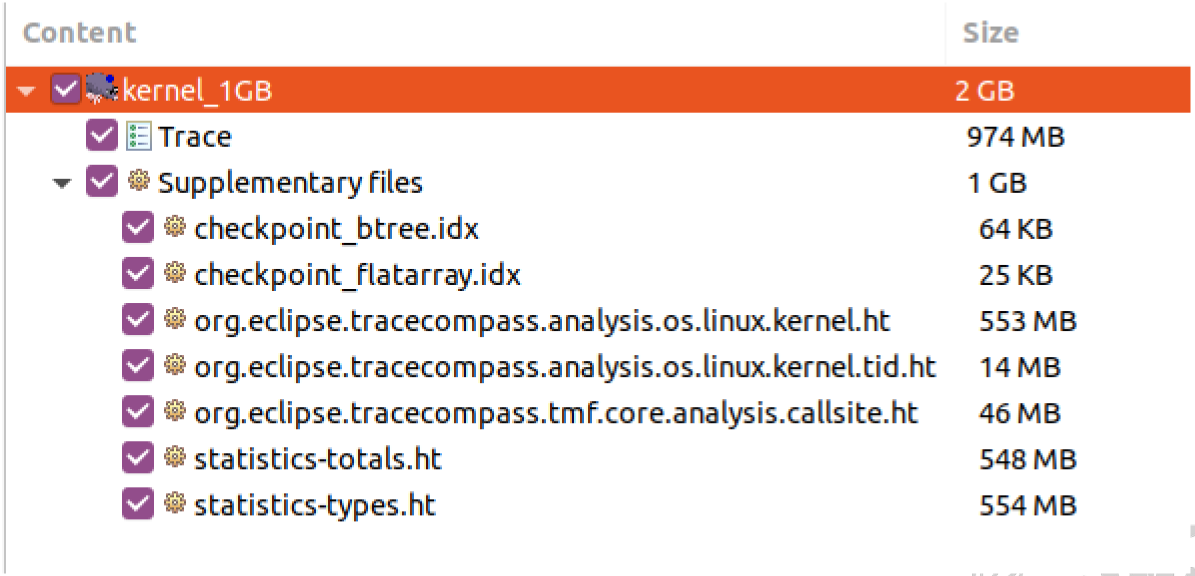
Motivation for this work

- Increased demand for analyzing very large traces

- History tree files are 1 to 2 times bigger than the trace itself

- Reduce disk space usage and files transfer time

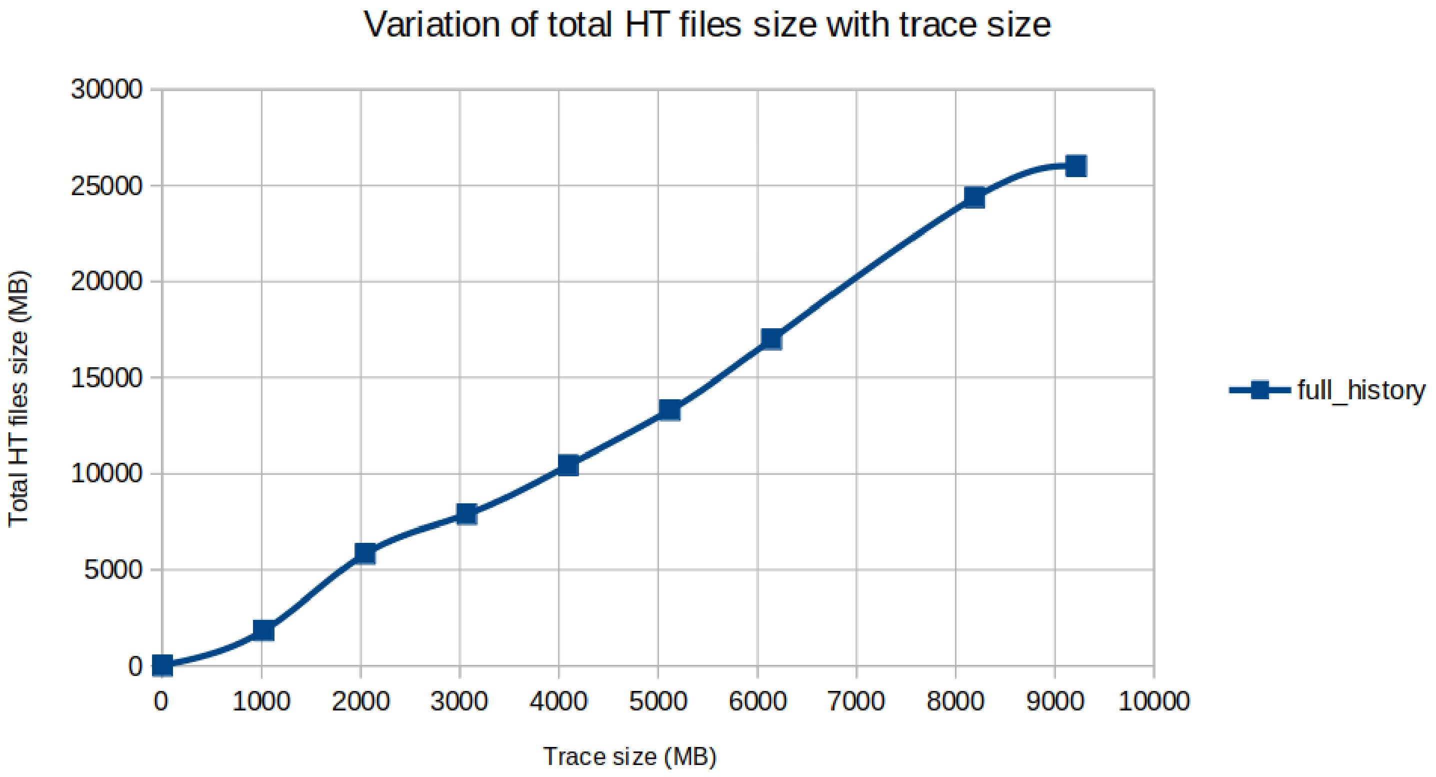
Trace size: 1GB
HT files total size: 1,8 GB



Content	Size
☑ kernel_1GB	2 GB
☑ Trace	974 MB
☑ Supplementary files	1 GB
☑ checkpoint_btree.idx	64 KB
☑ checkpoint_flatarray.idx	25 KB
☑ org.eclipse.tracecompass.analysis.os.linux.kernel.ht	553 MB
☑ org.eclipse.tracecompass.analysis.os.linux.kernel.tid.ht	14 MB
☑ org.eclipse.tracecompass.tmf.core.analysis.callsite.ht	46 MB
☑ statistics-totals.ht	548 MB
☑ statistics-types.ht	554 MB

Full History tree files for 1GB trace

Motivation for this work



What is the partial history tree ?

- Saving the complete state only at specific checkpoints

- Checkpoints selected at regular event intervals or possibly regular time intervals

- Recover the missing states at query time by reading the trace

What is the partial history ?

Fig.1: Intervals forming a history

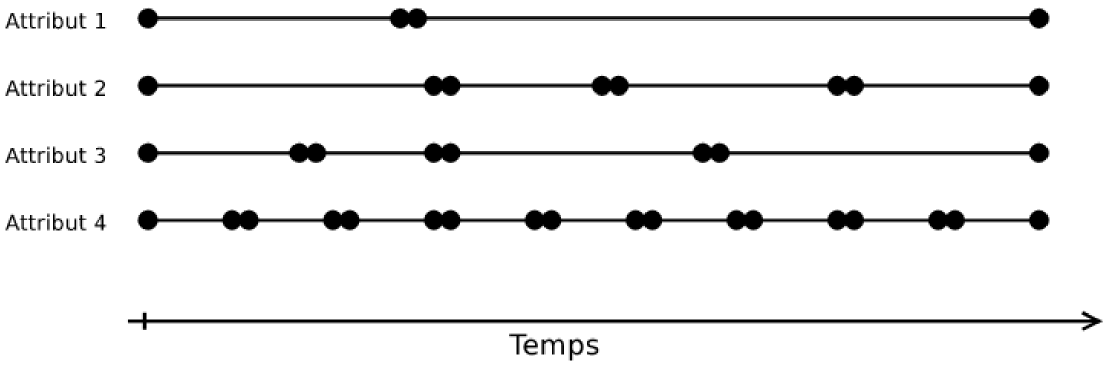
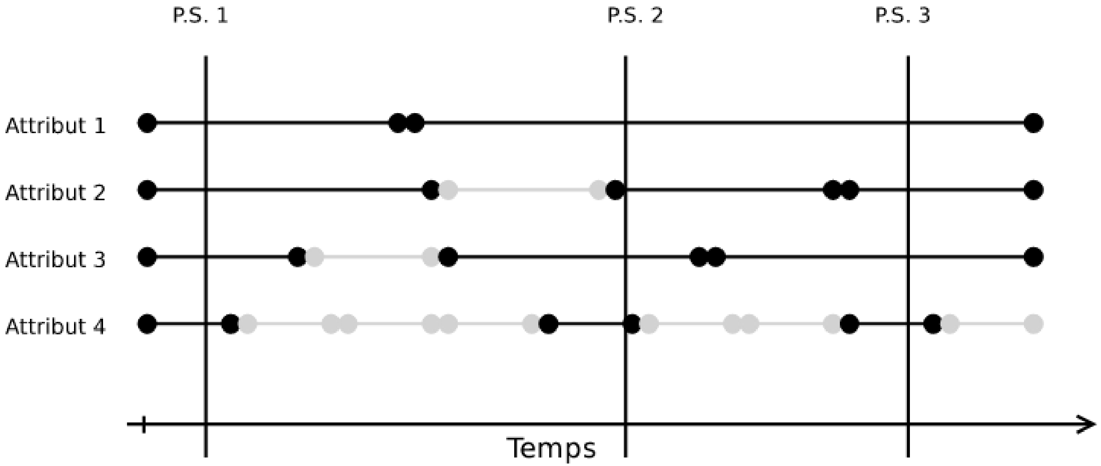


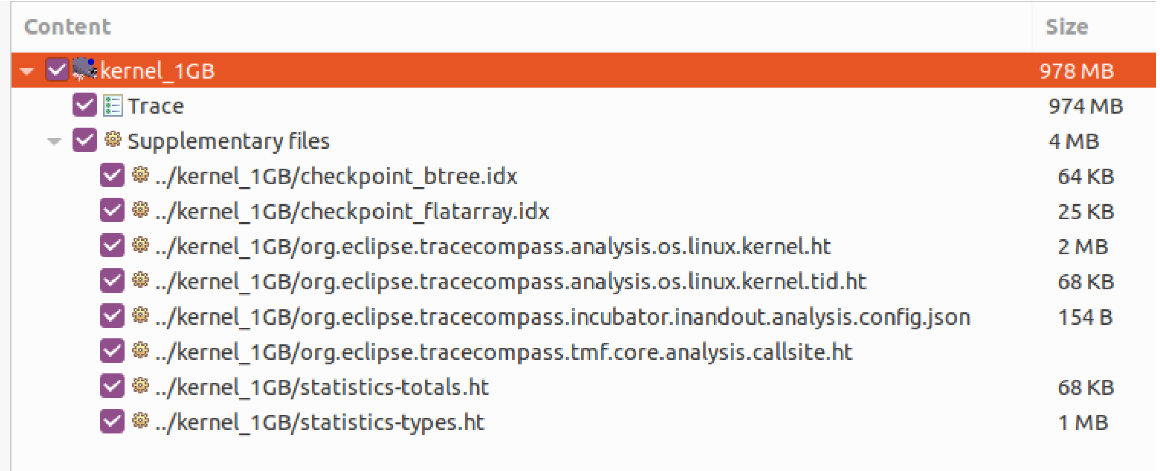
Fig.2: Not storing intervals that do not intersect checkpoints



A. Montplaisir, “Stockage sur disque pour acces rapide d'attributs avec intervalles de temps,” Master's thesis, École Polytechnique de Montréal, Dec. 2011.

Results: partial history tree files sizes

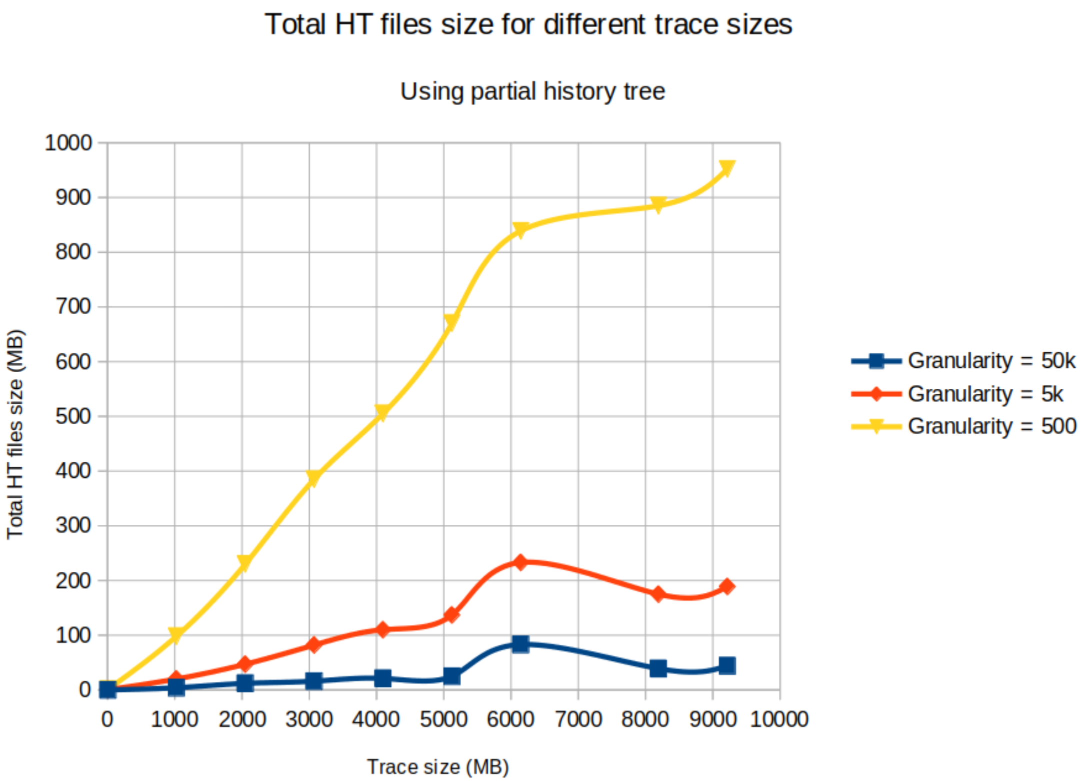
Trace size: 1GB
HT files total size: 4 MB



Content	Size
☑ kernel_1GB	978 MB
☑ Trace	974 MB
☑ Supplementary files	4 MB
☑ ../kernel_1GB/checkpoint_btree.idx	64 KB
☑ ../kernel_1GB/checkpoint_flatarray.idx	25 KB
☑ ../kernel_1GB/org.eclipse.tracecompass.analysis.os.linux.kernel.ht	2 MB
☑ ../kernel_1GB/org.eclipse.tracecompass.analysis.os.linux.kernel.tid.ht	68 KB
☑ ../kernel_1GB/org.eclipse.tracecompass.incubator.inandout.analysis.config.json	154 B
☑ ../kernel_1GB/org.eclipse.tracecompass.tmf.core.analysis.callsite.ht	
☑ ../kernel_1GB/statistics-totals.ht	68 KB
☑ ../kernel_1GB/statistics-types.ht	1 MB

Partial History tree files for 1GB trace (granularity = 50k)

Results: partial history tree files sizes



However, the greater the granularity, the slower the query that will populate the view, a trade-off should be made.

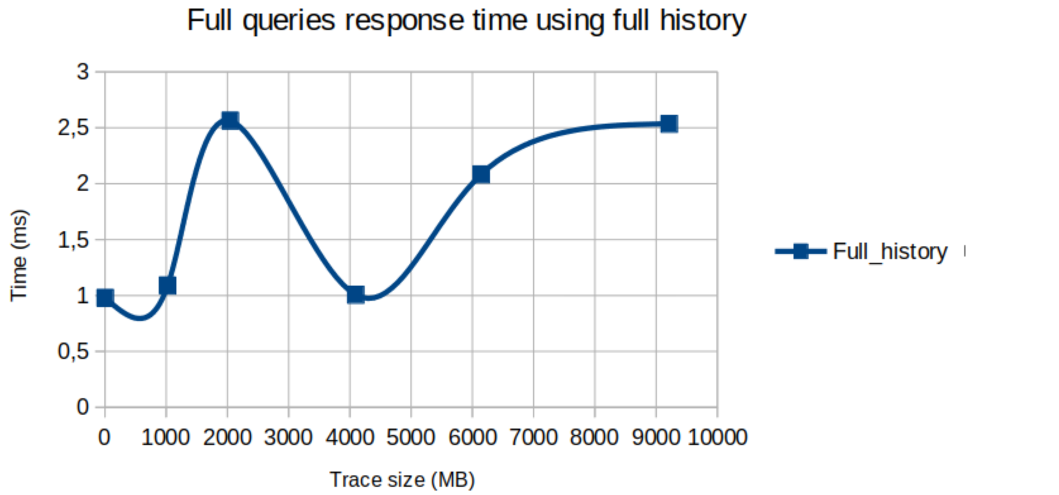
The initial prototype of the partial history tree

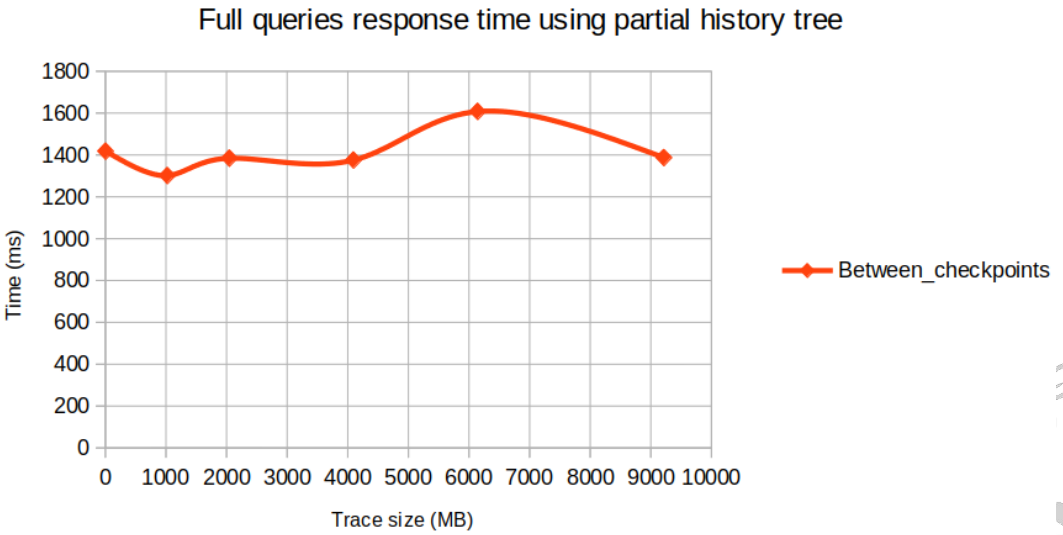
- Not all the analysis are available: only statistics analysis
- Missing intervals end times
- Uses a constant granularity (fixed to 50k events)
- Not all the queries are available

Contributions & improvements

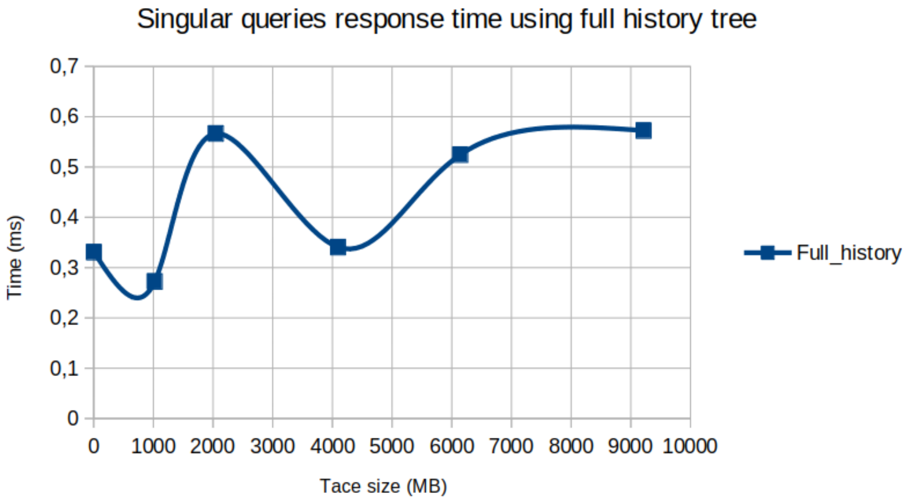
- Initial implementation of query2D() and doSingularQuery() for the partial history tree
- Enhanced doQuery() : uses real intervals end times
- Views populated with states, ex: control flow /resources view

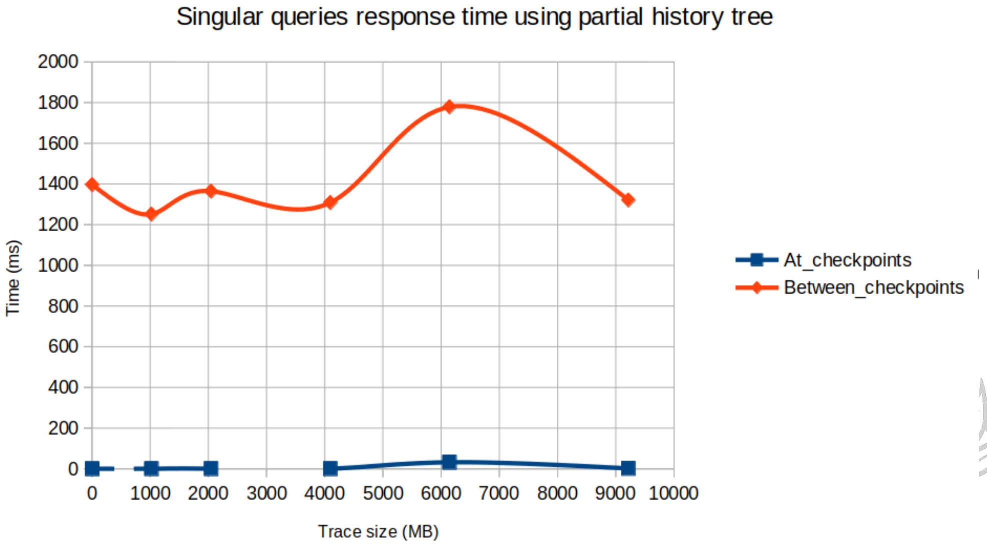
Results: queries response times





Results: queries response times





Results: queries response times

- Early results subject to verification and optimization
- The response time for the queries used with the partial history are pretty high
- Queries will speed-up by synchronizing the timestamps displayed on an analysis view with the checkpoints

Work in progress

- Improve the performance of the range queries.
- Select checkpoints based on time intervals
- Adaptive interval between the checkpoints

Conclusion

- Reduction of the history tree files size
- Exporting pre-processed trace data becomes quicker even with large traces
- Study the performance of Trace Compass on these huge traces and further optimize
- Keep Trace Compass the best tool for huge traces!

Q&A

Email: abdellah.rahmani@polymtl.ca