Excel@FIT 2025

BRNO UNIVERSITY OF TECHNOLOGY FACULTY OF INFORMATION TECHNOLOGY

EdgeTrace: Automated Call Graph Comparison for GraalVM Native Image

AUTHOR
Milan Vodák

SUPERVISOR
Ing. David Kozák

Native Image: Java to Native Binaries

GraalVM Native Image produces **native executables** from Java bytecode – no JVM required at runtime.

How to minify the executable? **Points-to analysis** (PTA) discovers reachable program elements to compile [2].

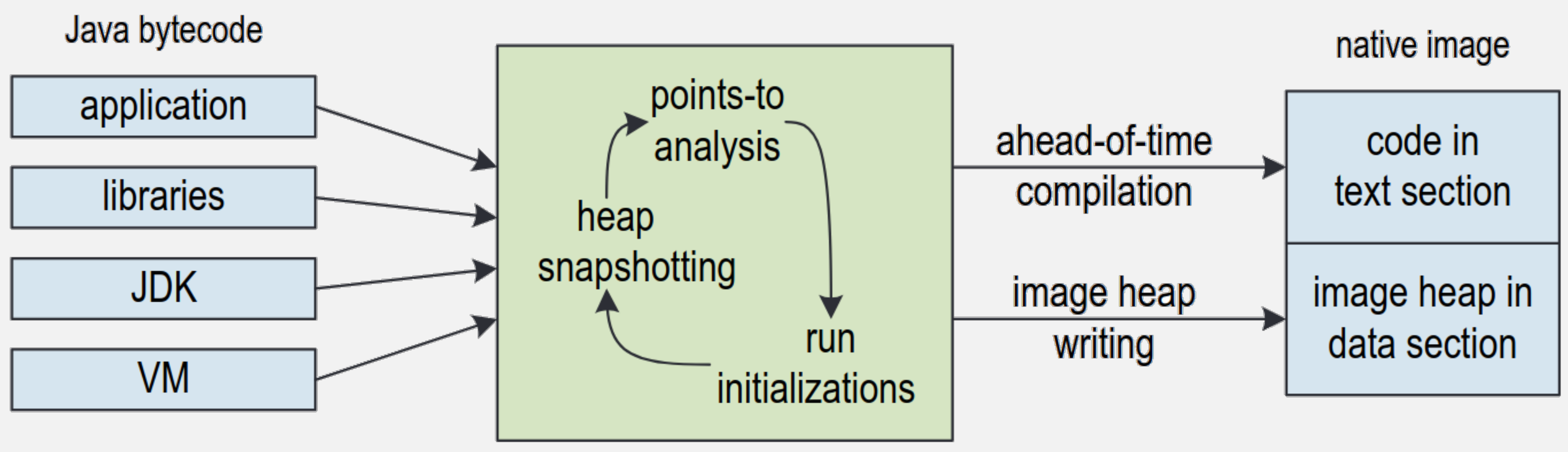


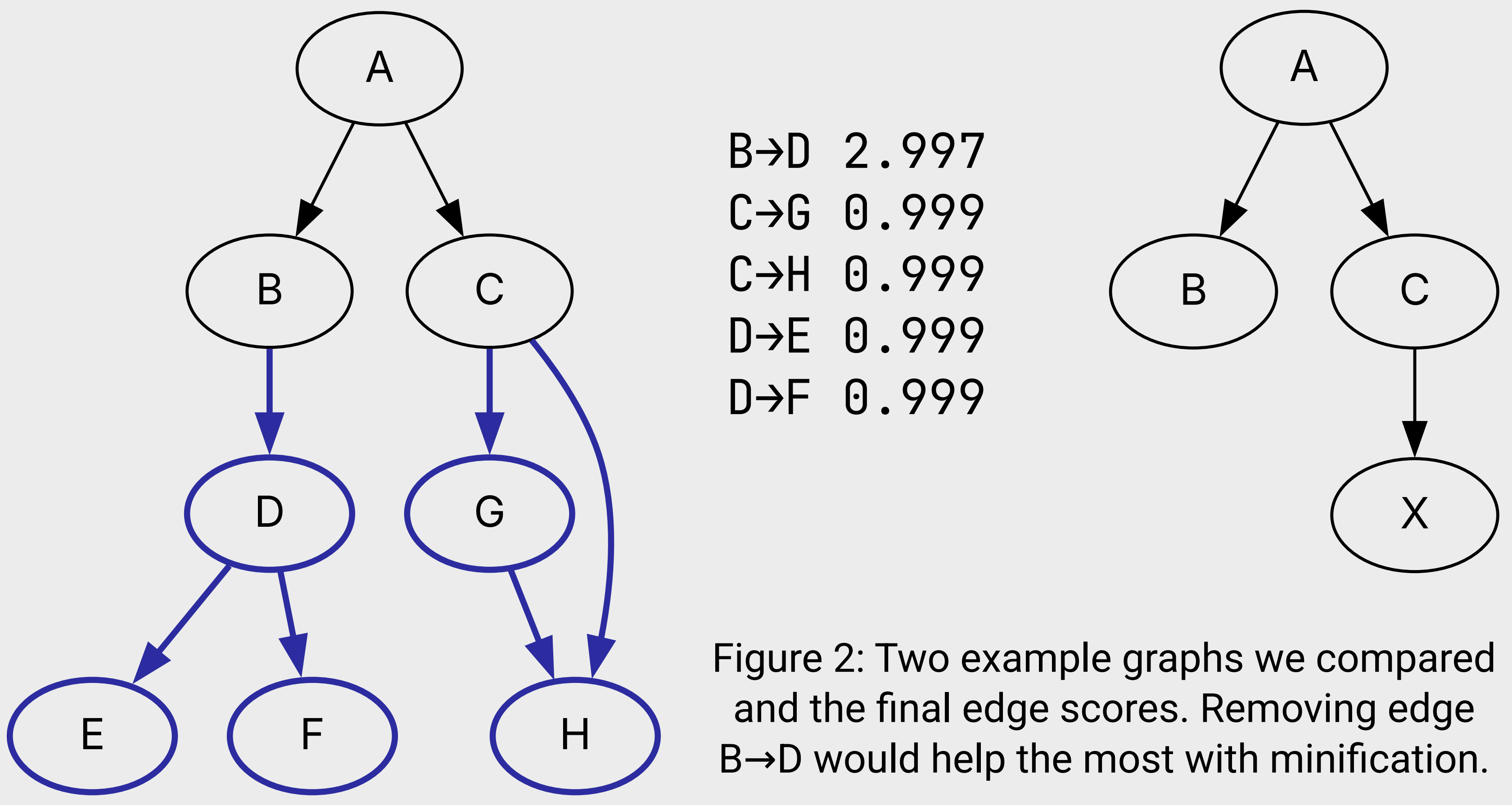
Figure 1: Steps that Native Image performs at build time. Taken from [2].

Static analysis of large applications is **challenging**. We had **no tool to track impacts** of PTA changes.

How to evaluate precision of Native Image static analysis? By comparing call graphs.

Comparing Call Graphs

How to compare call graphs? We want to **find edges** from the larger graph that cause the most differences.



```
B→D 2.997
C→G 0.999
C→H 0.999
D→E 0.999
D→F 0.999
```

Figure 2: Two example graphs we compared and the final edge scores. Removing edge B→D would help the most with minification.

The algorithm by Lhotak et al. [1] simulates the flow of a fluid in reverse **from methods to entrypoints**. **More fluid** flowing through an edge → **higher score**.

We use this algorithm to find edges that are most responsible for the difference.

Call Graphs Visualization

EdgeTrace is a web application that:

- allows import of Native Image reports,
- displays call graphs in an interactive way,
- shows the most important differences.

Key features:

- **Neo4j** graph database to save call graphs,
- Cypher queries to find methods and paths,
- visualization using **Cytoscape.js** frontend,
- the algorithm runs via **Python–C** bindings.

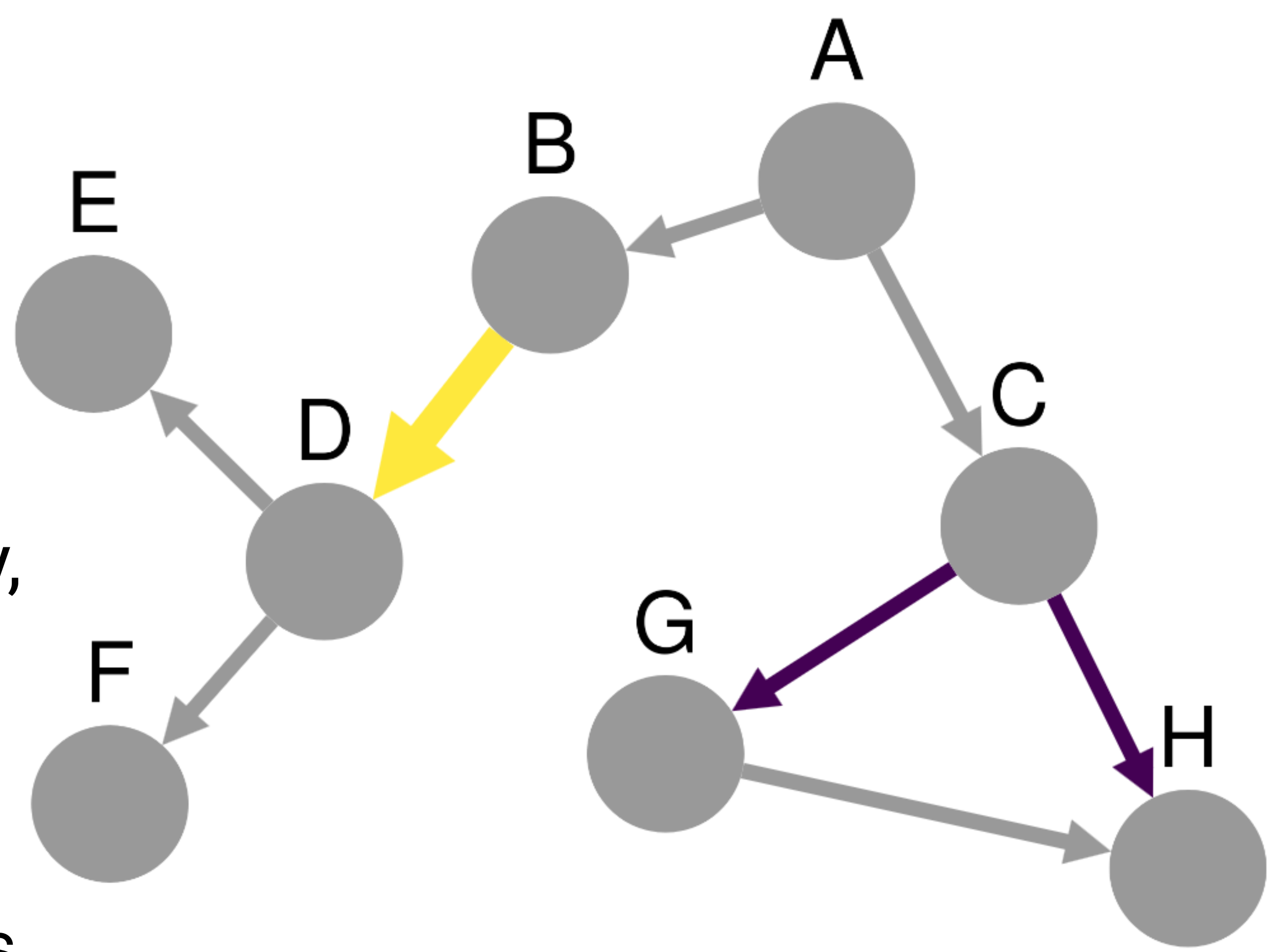


Figure 3: Call graph from Figure 2 with difference visualized in EdgeTrace.

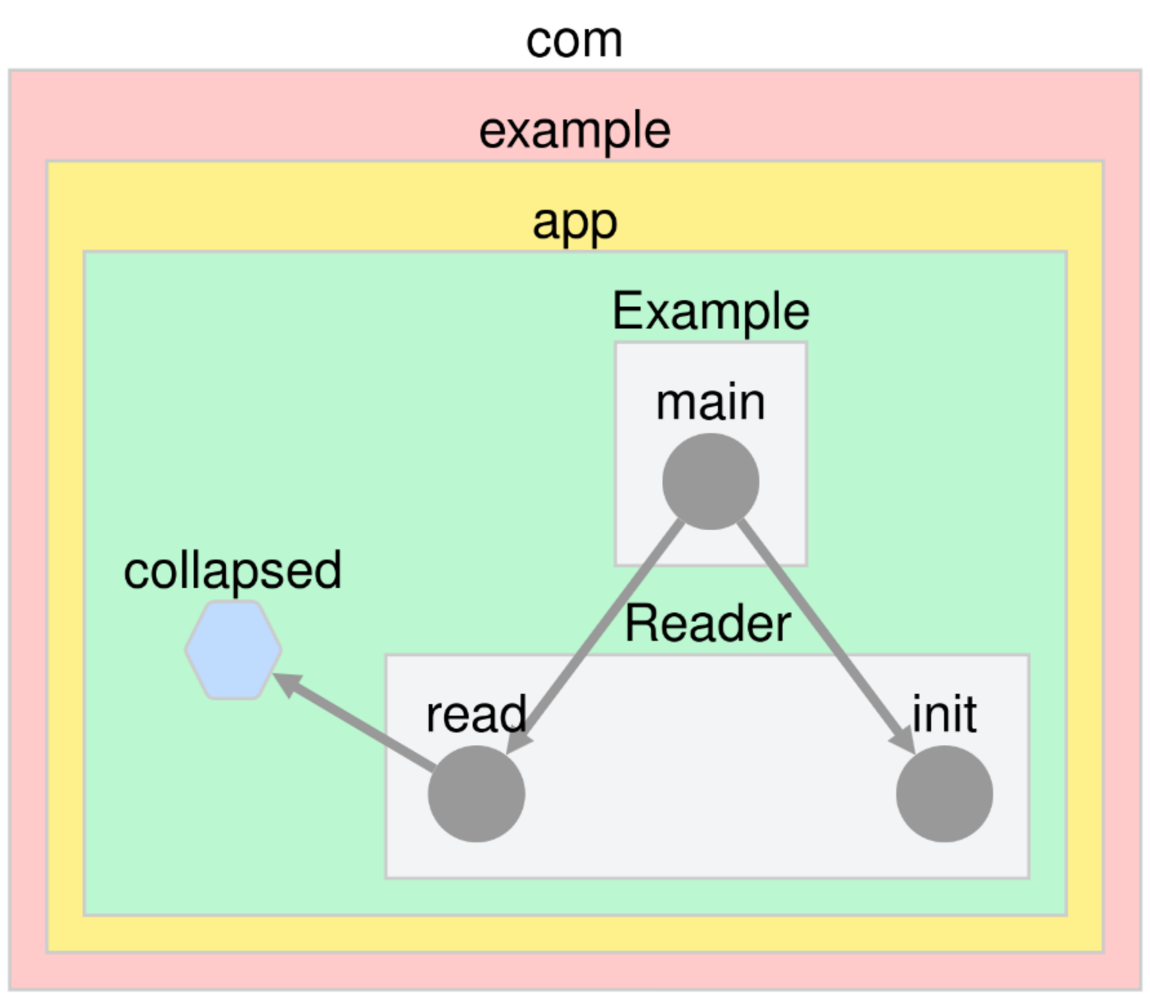
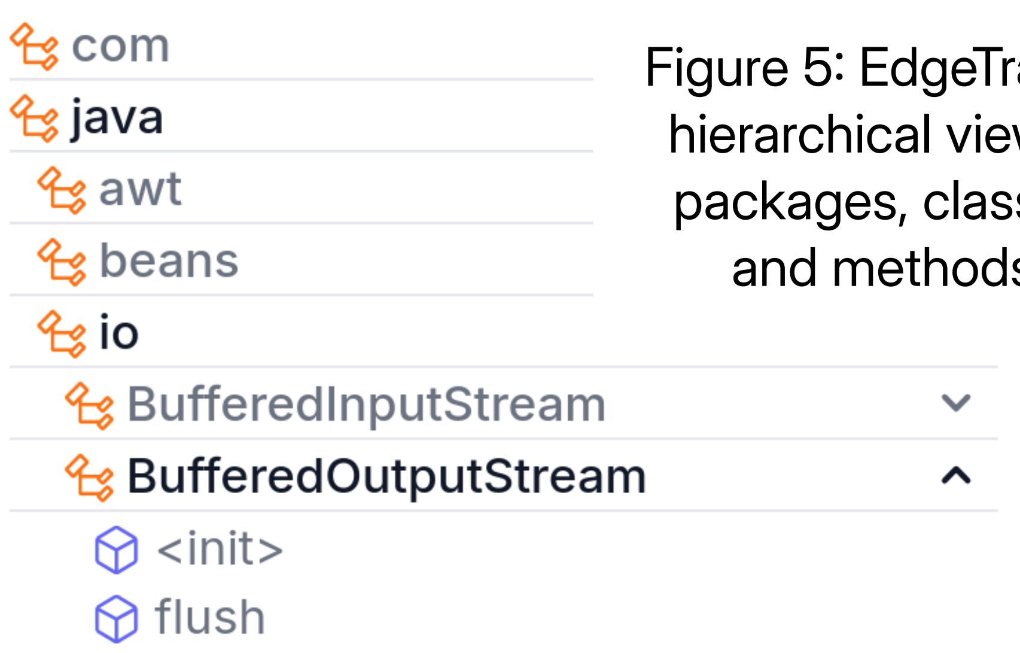


Figure 4: A call graph in EdgeTrace with package and class compound nodes enabled.



com
java
awt
beans
io
BufferedInputStream
BufferedOutputStream
<init>
flush

Figure 5: EdgeTrace's hierarchical view of packages, classes, and methods.

EdgeTrace processes call graphs generated by Native Image, computes the difference and displays it in an intuitive way.

[1] Lhoták, O. Comparing call graphs. In: Proceedings of the 7th ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering. New York, NY, USA: Association for Computing Machinery, 2007, p. 37–42. PASTE '07. ISBN 9781595935953. Available at: https://doi.org/10.1145/1251535.1251542.

[2] Wimmer, C.; Stancu, C.; Hofer, P.; Jovanovic, V.; Wögerer, P. et al. Initialize once, start fast: application initialization at build time. Proc. ACM Program. Lang. New York, NY, USA: Association for Computing Machinery, october 2019, vol. 3, OOPSLA. Available at: https://doi.org/10.1145/3360610.