JOHANNES KEPLER UNIVERSITY LINZ

ssw.jku.at

Combining Execution Modes of LLVM Bitcode on GraalVM

ORACLE GraalVM

graalvm.org

Christoph Pichler
Johannes Kepler University
christoph.pichler@jku.at

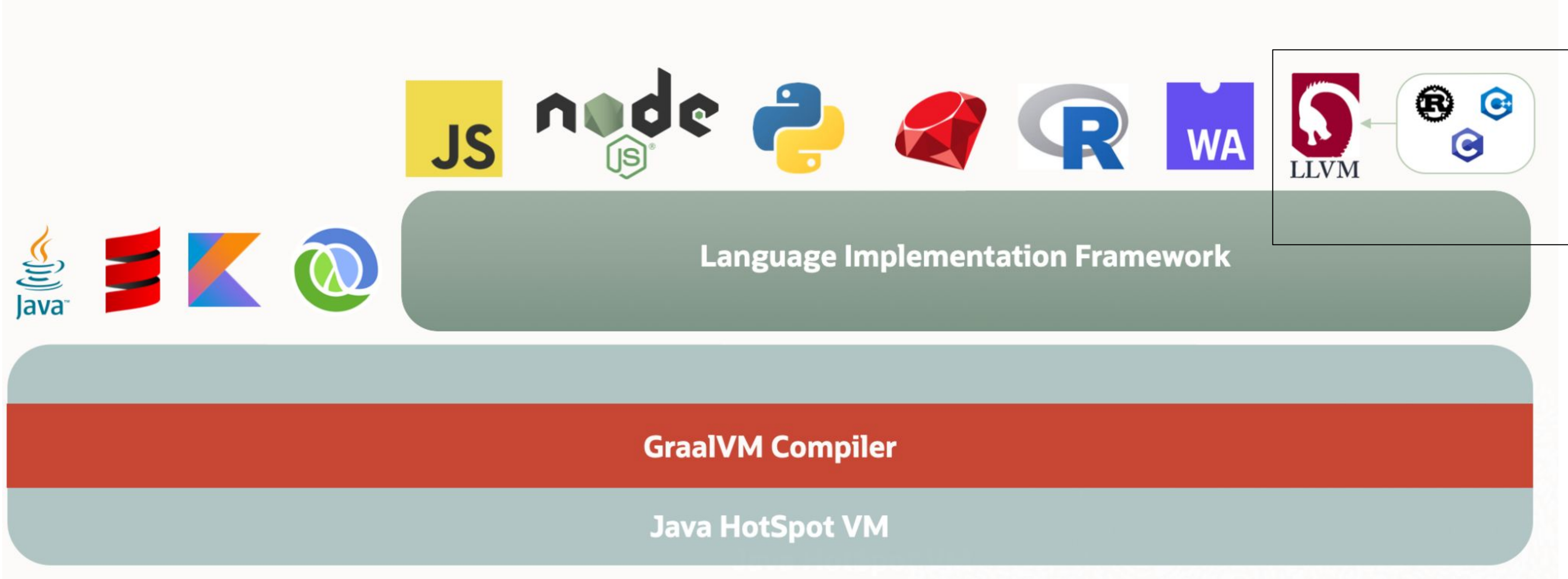
Paley Li
Oracle Labs
paley.li@oracle.com

Roland Schatz
Oracle Labs
roland.schatz@oracle.com

Hanspeter Mössenböck
Johannes Kepler University
hanspeter.moessenboeck@jku.at

GraalVM

- Polyglot virtual machine
  - No cross-language overhead
- Highly-optimizing JIT-compiler
  - Dynamic and speculative



➔ Execution of C code via clang + LLVM bitcode

Why Not Only Native Execution?

Python/JS/… pseudo code

```
p = Point(x: 4, y: -3)
diff = calcDiff(p)
```

passing managed (e.g. JS) object to C code

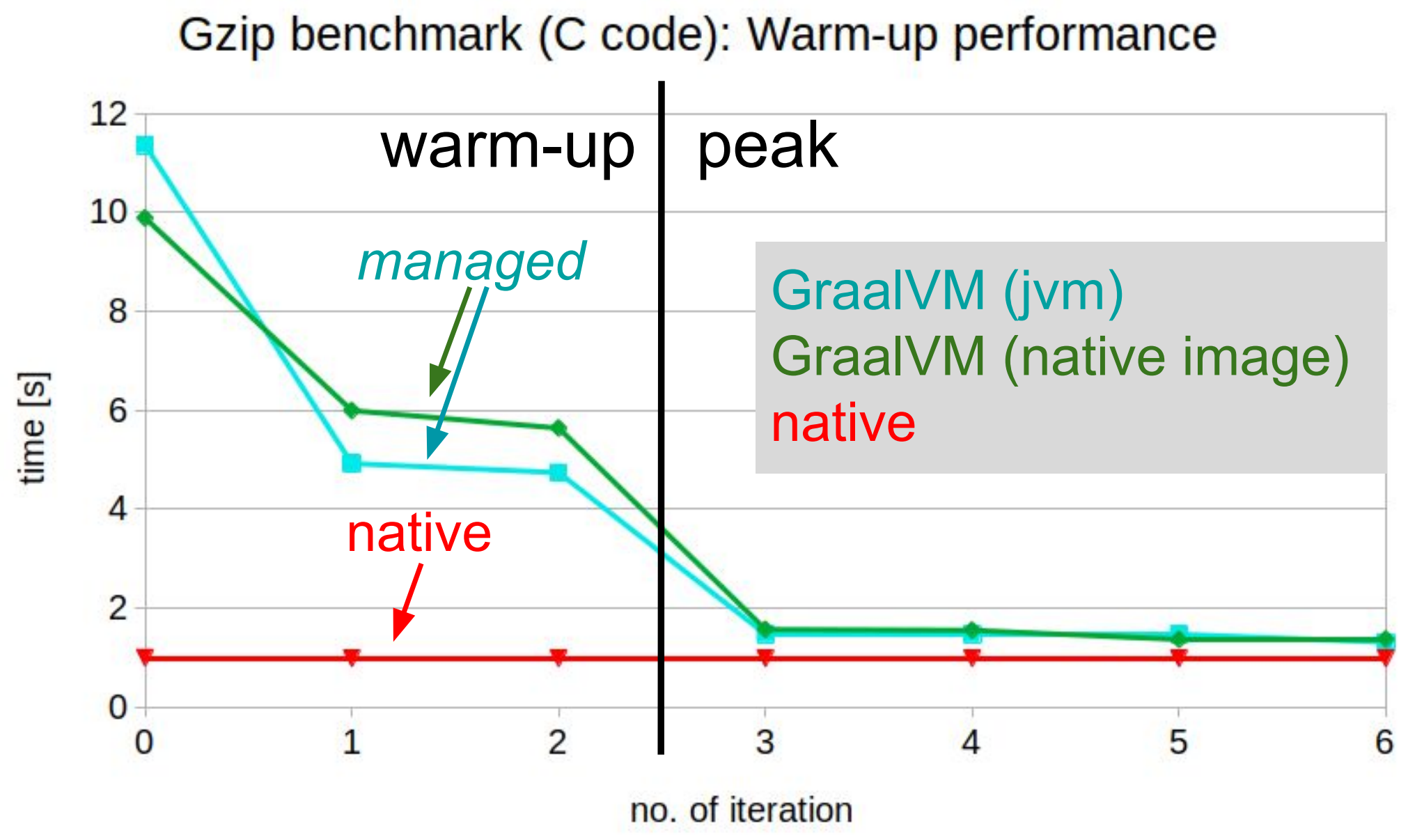
C code

```
double calcDiff(struct Point *p) {
  return sqrt(p->x*p->x+p->y*p->y);
}
```

"ERROR: managed object p not accessible in (natively executed) C code"

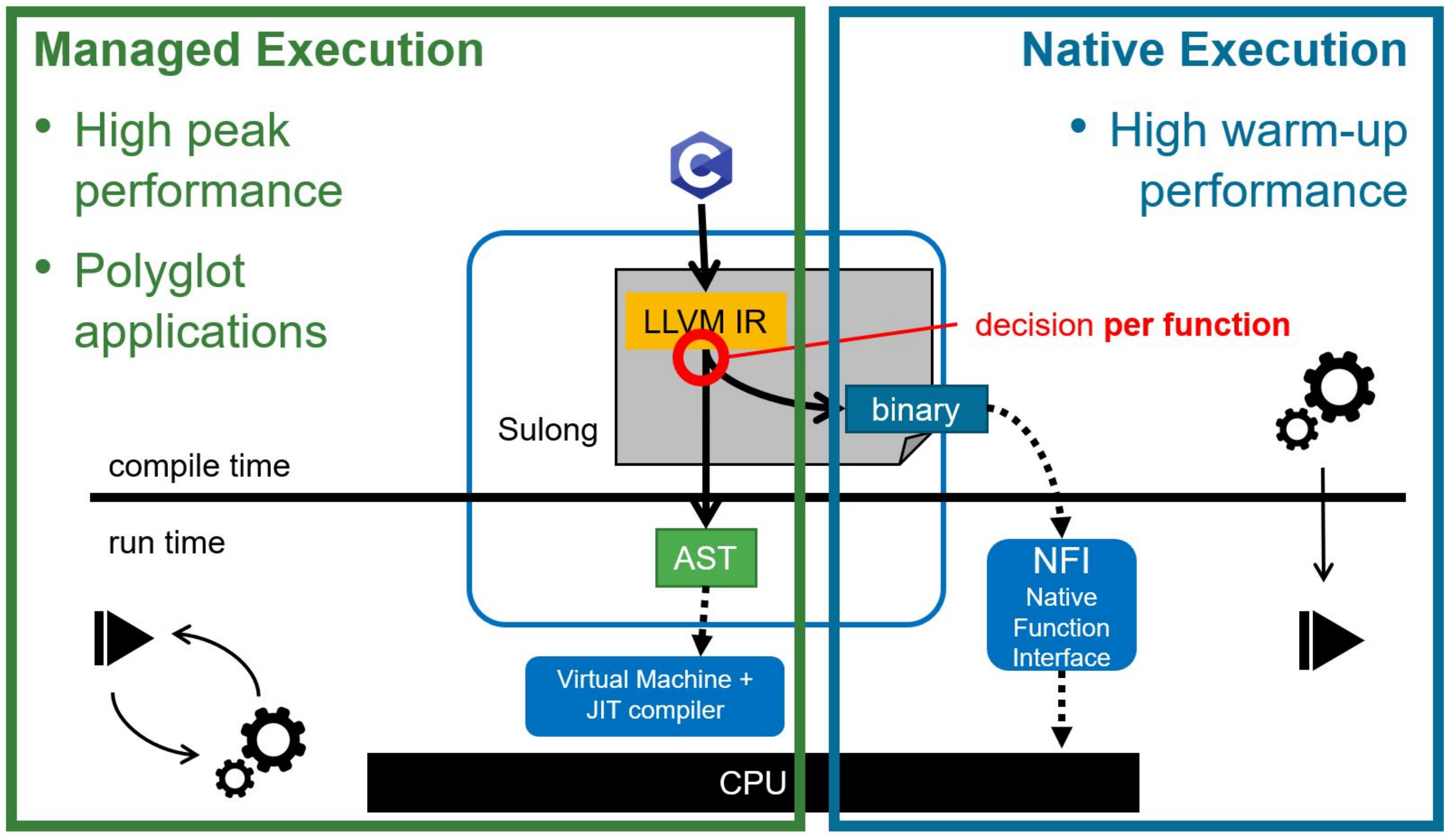
- Existing polyglot systems: Overhead for cross-language calls

Why Not Only Managed Execution?



- Slow warm-up for managed execution in GraalVM (due to dynamic compilation)
- Thus: Less managed/more native code improves (warm-up) performance

Our Approach: Hybrid Execution



Managed Execution

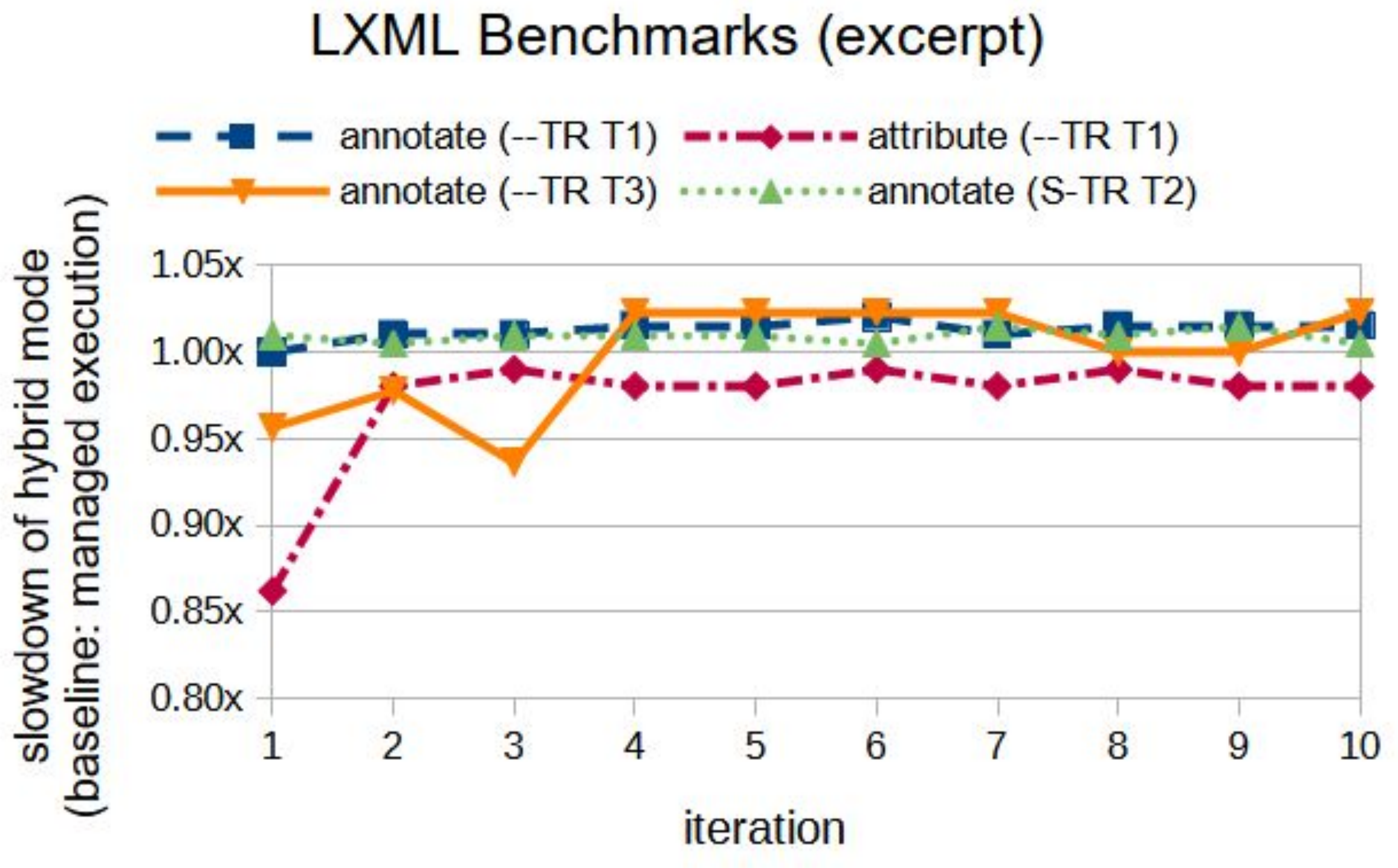
- High peak performance
- Polyglot applications

Native Execution

- High warm-up performance

Current State

- Benchmark: LXML parser in Python, which uses libxml in C
- Interquartile range: 0.02x



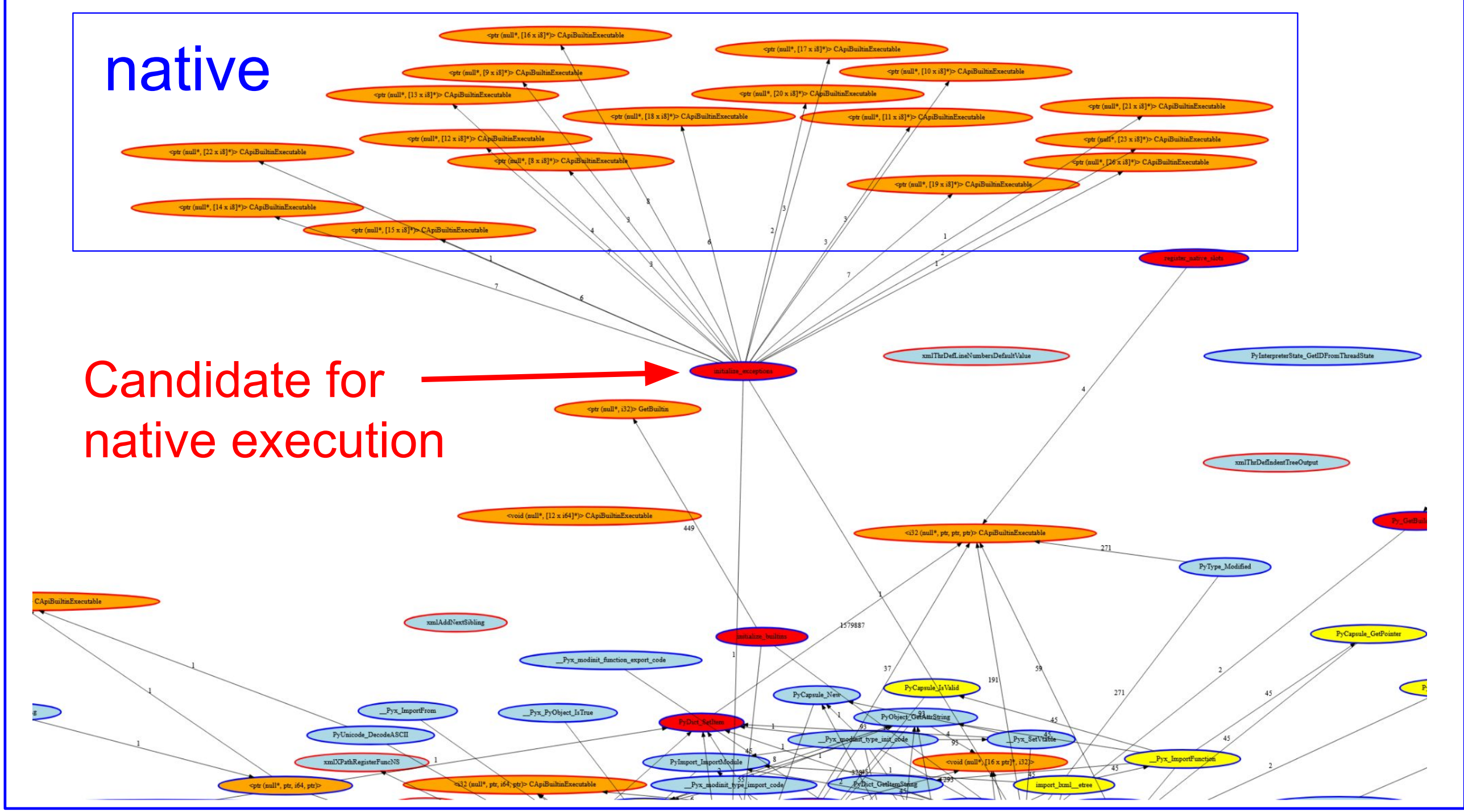
- Different tasks show different results
→ performance highly depends on decision which function to execute natively or via GraalVM
- Restrictions in current state
  - Manual decision how a callee is run (GraalVM/native)

Proof of Concept

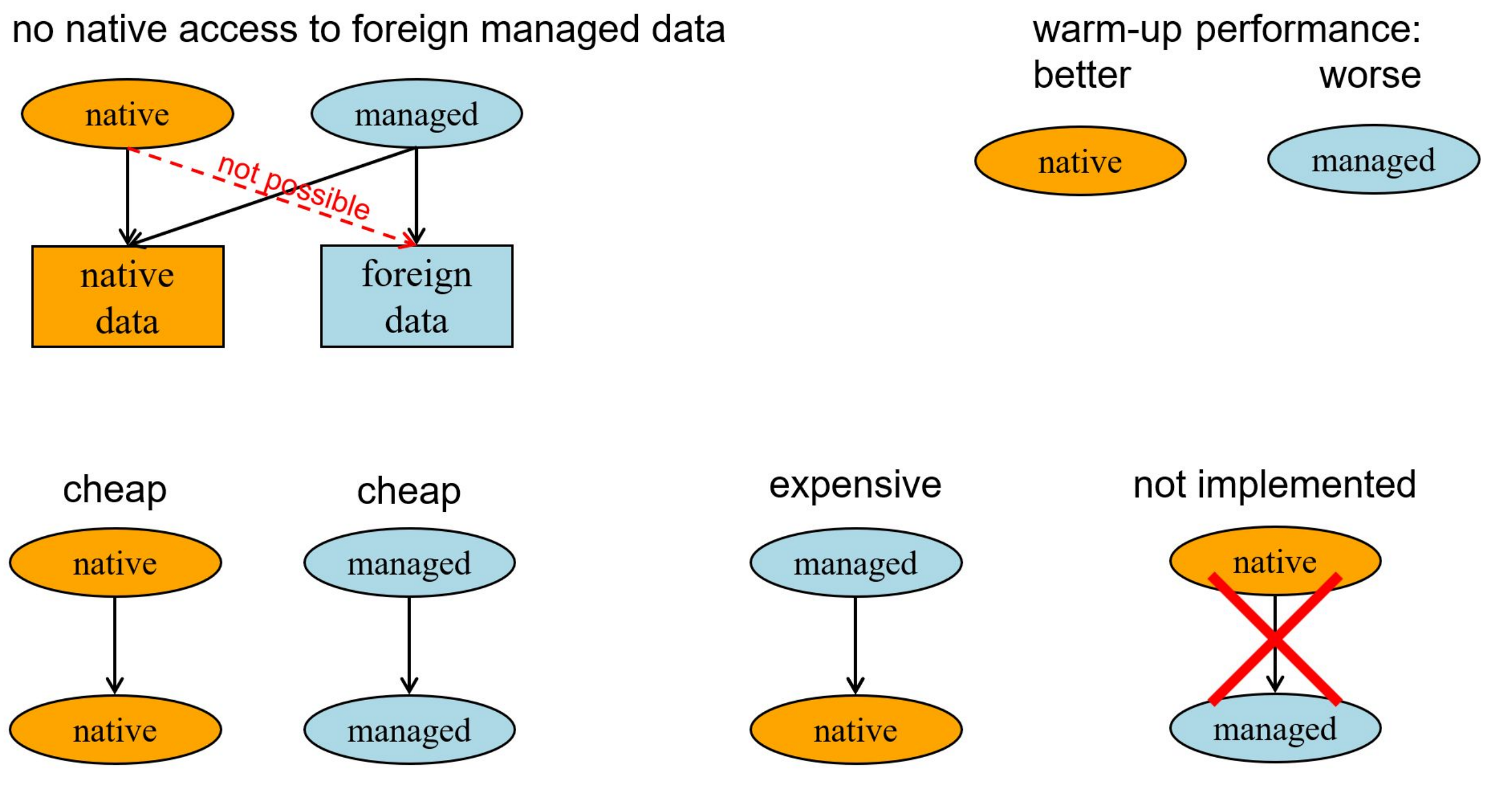
Christoph Pichler, Paley Li, Roland Schatz, and Hanspeter Mössenböck: Hybrid Execution: Combining Ahead-of-Time and Just-in-Time Compilation. VMIL 2023, Cascais, Portugal. https://doi.org/10.1145/3623507.3623554

Next Step: Automatically Decide Execution Mode

Create a Call Graph from the Application



Find Heuristic to Decide Execution Mode



References

- graalvm.org
- Christoph Pichler, Paley Li, Roland Schatz, and Hanspeter Mössenböck. 2023. Hybrid Execution: Combining Ahead-of-Time and Just-in-Time Compilation. In Proceedings of the 15th ACM SIGPLAN International Workshop on Virtual Machines and Intermediate Languages (VMIL 2023). Association for Computing Machinery, New York, NY, USA, 39–49. https://doi.org/10.1145/3623507.3623554
- Thomas Würthinger, Christian Wimmer, Andreas Wöß, Lukas Stadler, Gilles Duboscq, Christian Humer, Gregor Richards, Doug Simon, and Mario Wolczko. 2013. One VM to rule them all. In Proceedings of the 2013 ACM international symposium on New ideas, new paradigms, and reflections on programming & software (Onward! 2013). Association for Computing Machinery, New York, NY, USA, 187–204. https://doi.org/10.1145/2509578.2509581
- Manuel Rigger, Matthias Grimmer, Christian Wimmer, Thomas Würthinger, and Hanspeter Mössenböck. 2016. Bringing low-level languages to the JVM: efficient execution of LLVM IR on Truffle. In Proceedings of the 8th International Workshop on Virtual Machines and Intermediate Languages (VMIL 2016). Association for Computing Machinery, New York, NY, USA, 6–15. https://doi.org/10.1145/2998415.2998416
- Manuel Rigger, Roland Schatz, Jacob Kreindl, Christian Häubl, and Hanspeter Mössenböck. 2018. Sulong, and thanks for all the fish. In Companion Proceedings of the 2nd International Conference on the Art, Science, and Engineering of Programming (Programming '18). Association for Computing Machinery, New York, NY, USA, 58–60. https://doi.org/10.1145/3191697.3191726
- T. Pittman. 1987. Two-level hybrid interpreter/native code execution for combined space-time program efficiency. In Papers of the Symposium on Interpreters and interpretive techniques (SIGPLAN '87). Association for Computing Machinery, New York, NY, USA, 150–152. https://doi.org/10.1145/29650.29666
- Manel Grichi, Mouna Abidi, Yann-Gaël Guéhéneuc, and Foutse Khomh. 2019. State of practices of Java native interface. In Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering (CASCON '19). IBM Corp., USA, 274–283.
- Matthias Grimmer, Roland Schatz, Chris Seaton, Thomas Würthinger, Mikel Luján, and Hanspeter Mössenböck. 2018. Cross-Language Interoperability in a Multi-Language Runtime. ACM Trans. Program. Lang. Syst. 40, 2, Article 8 (June 2018), 43 pages. https://doi.org/10.1145/3201898