Optimising the Mobile Net

Hugh Leather
Stephen Kyle, Volker Seeker

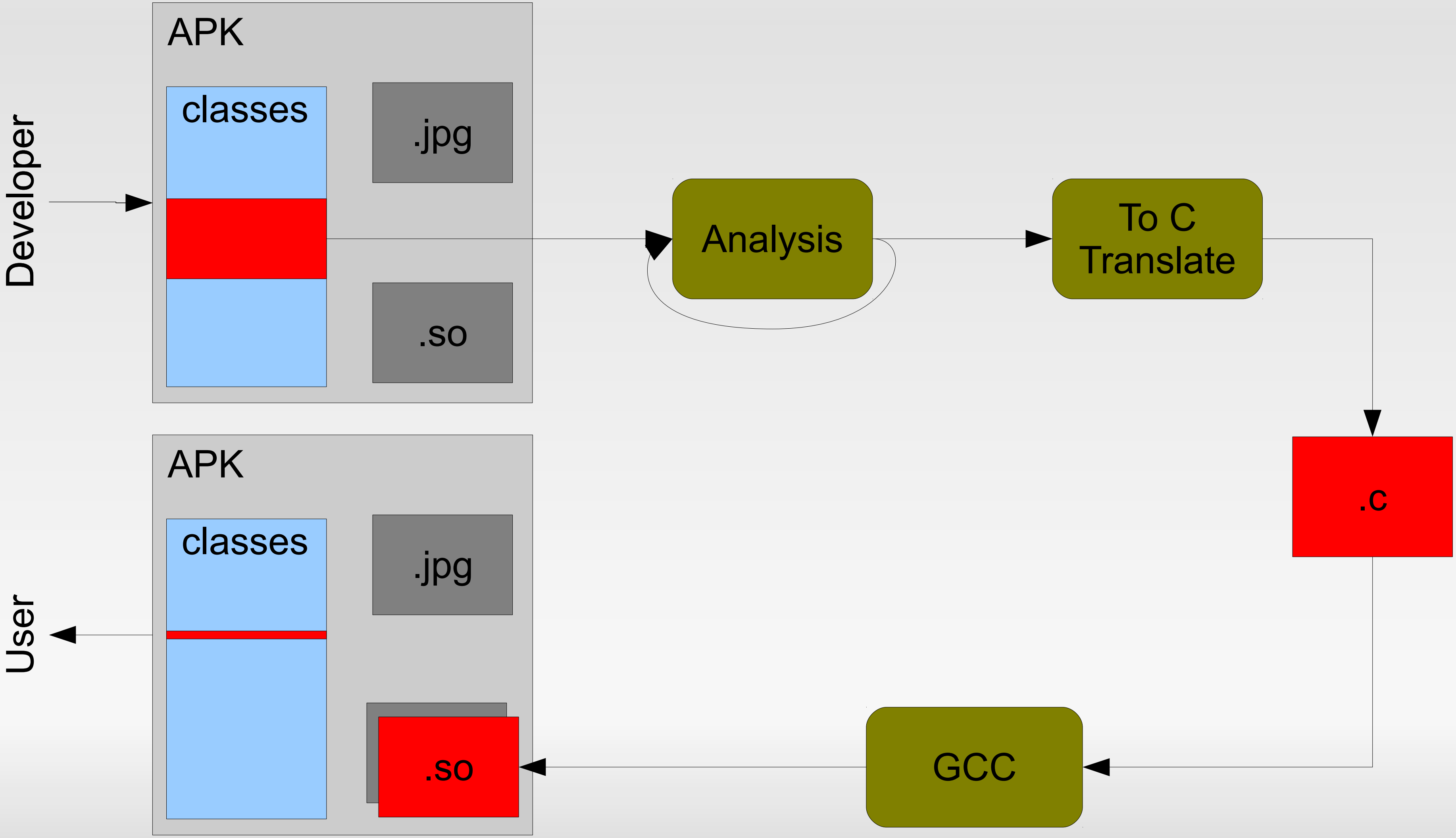
Compilers and Operating Systems

- Mobile devices are resource constrained
- All Android programs use Dalvik JIT - very slow
- Operating system does not mobile optimised

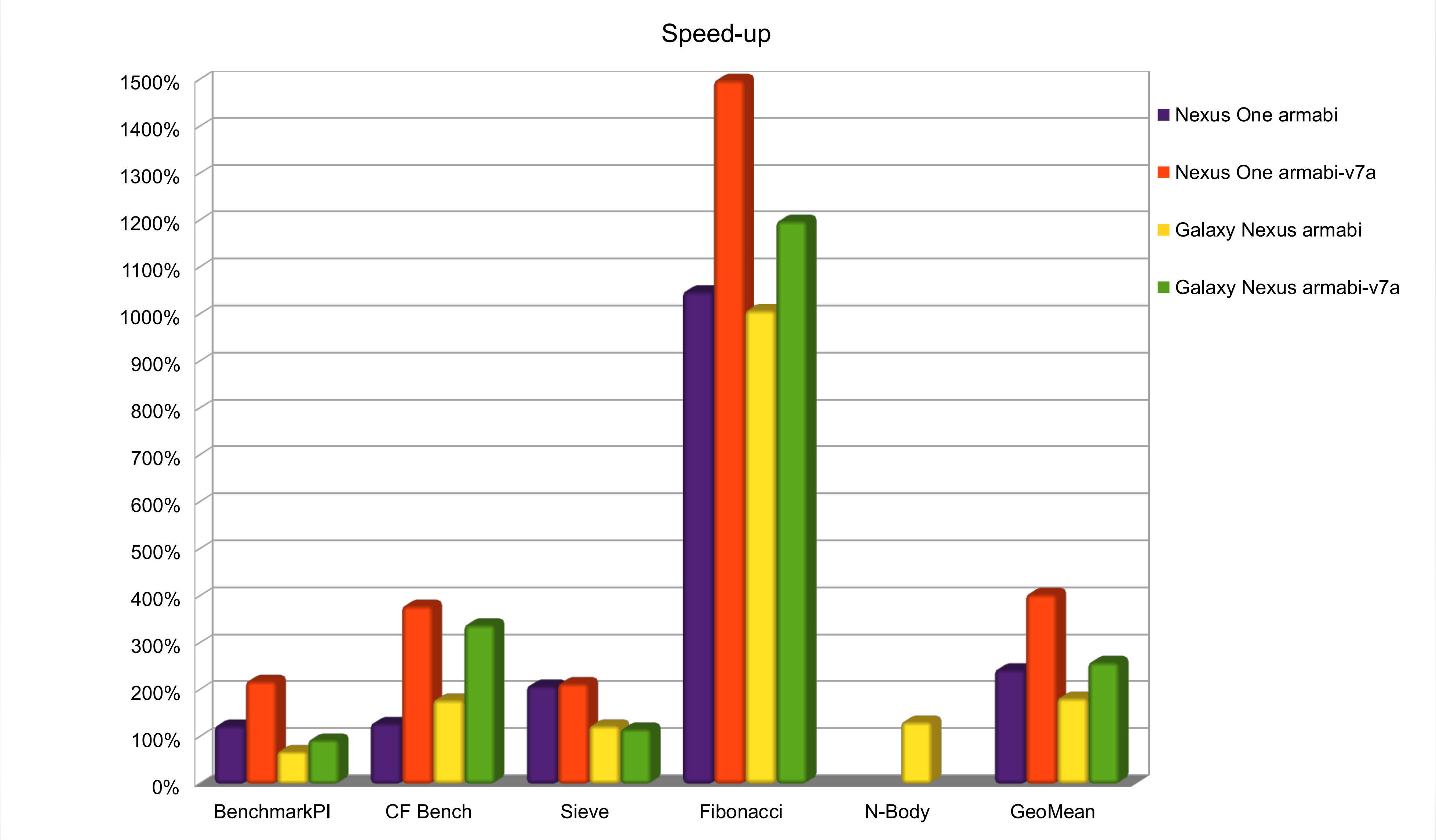
- Need to make them faster and lower power

Server-side Dalvik to Native

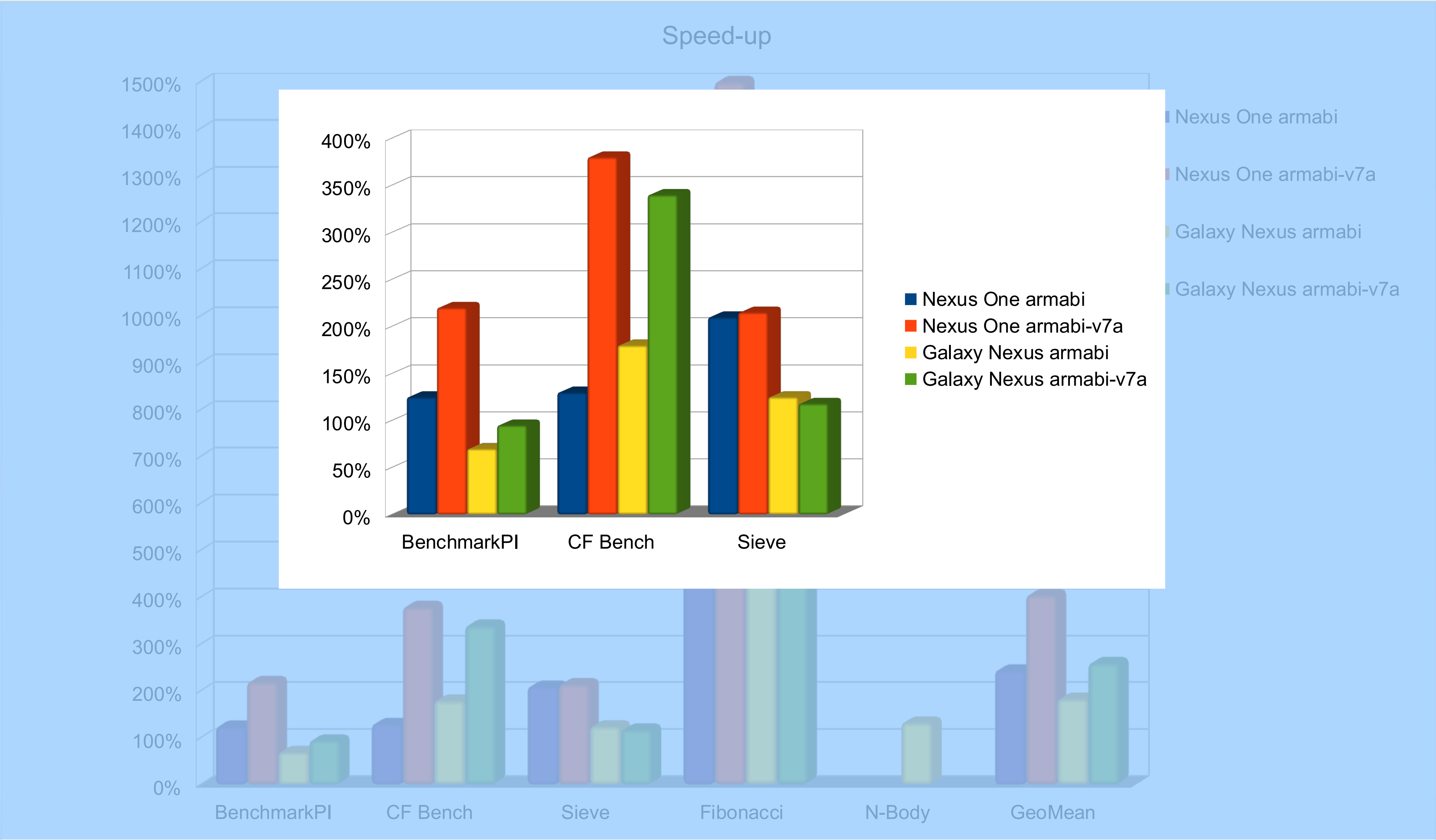
- Server side native compilation of hot methods



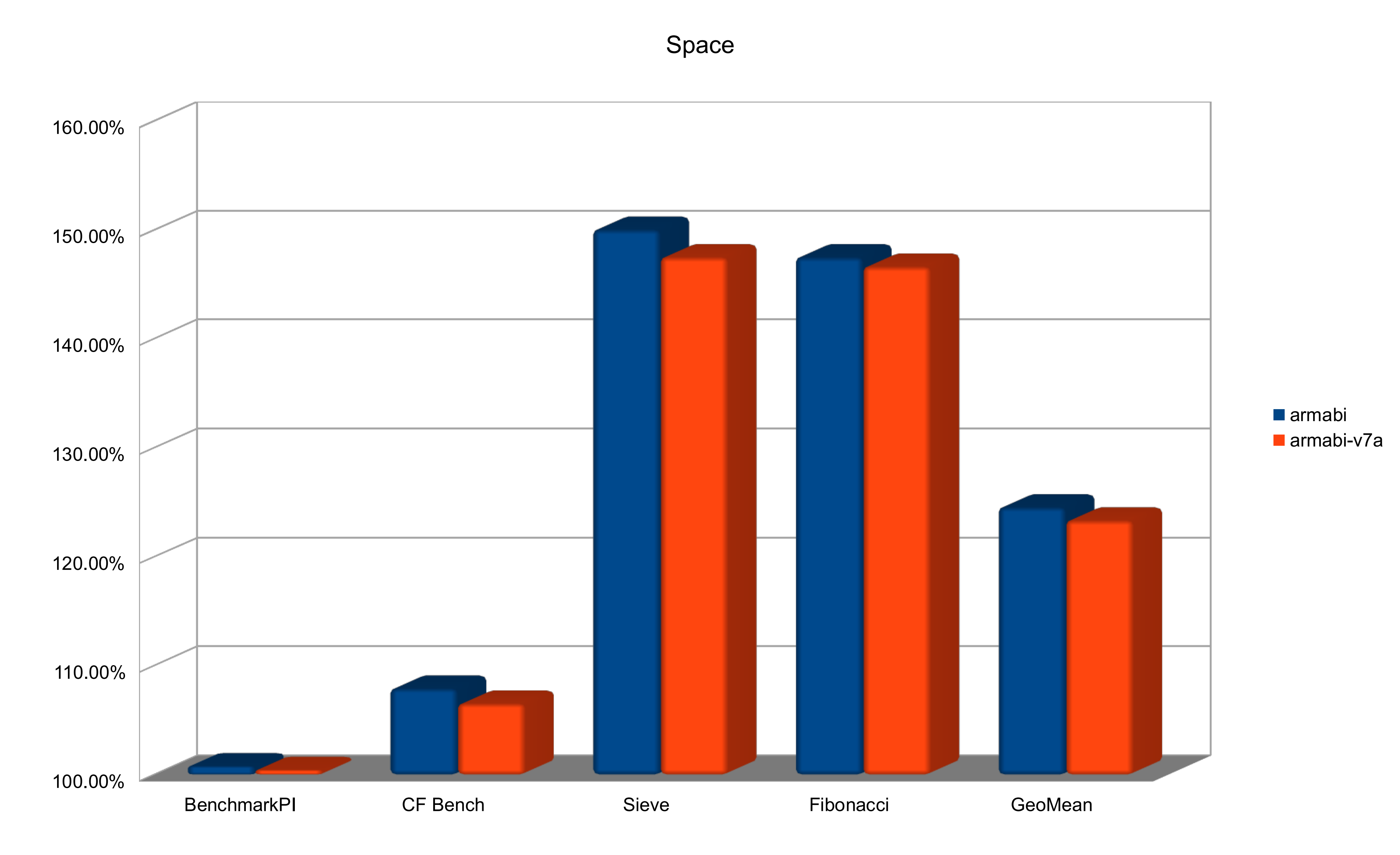
Server-side Dalvik to Native



Server-side Dalvik to Native

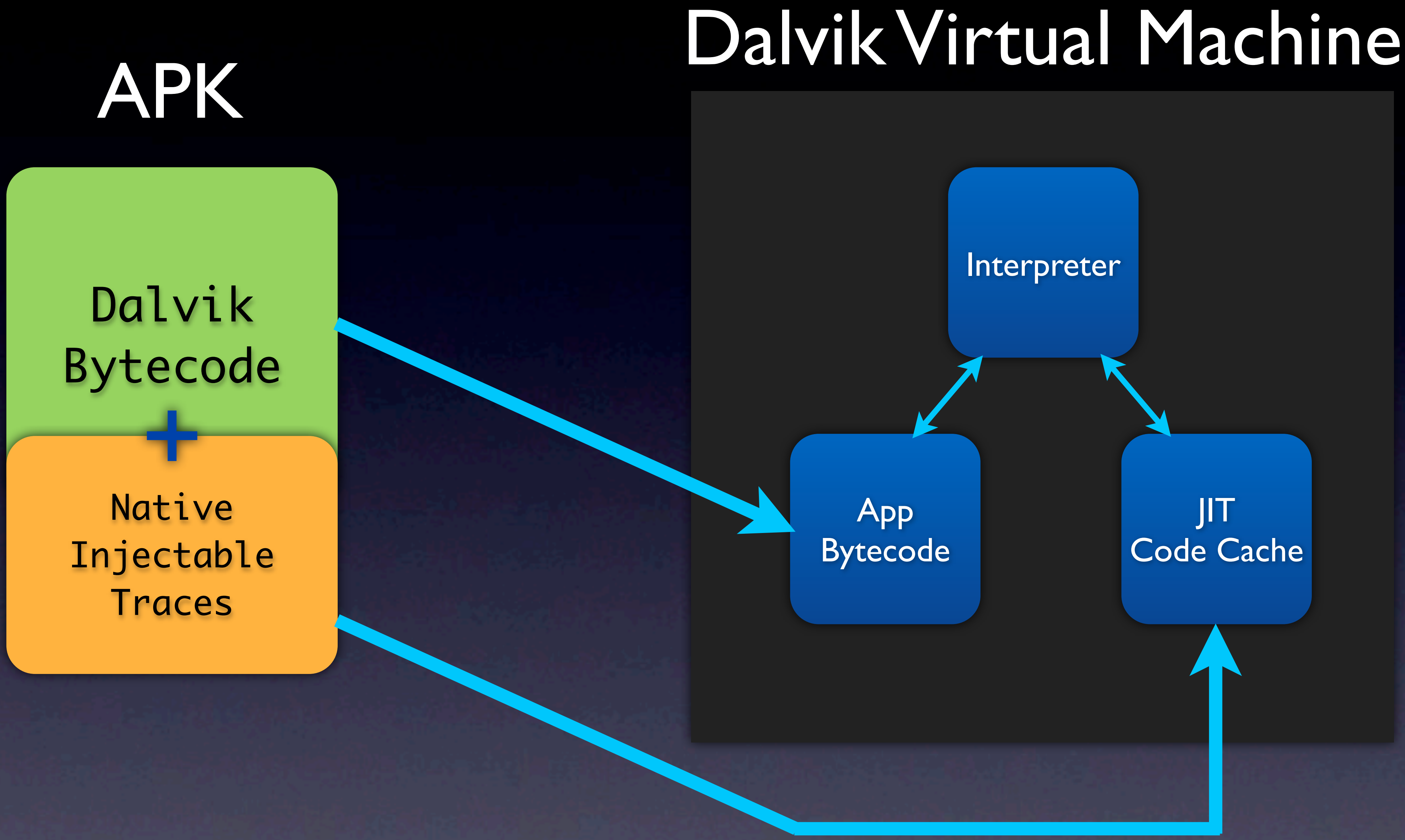


Server-side Dalvik to Native



Trace Injection

Stephen Kyle



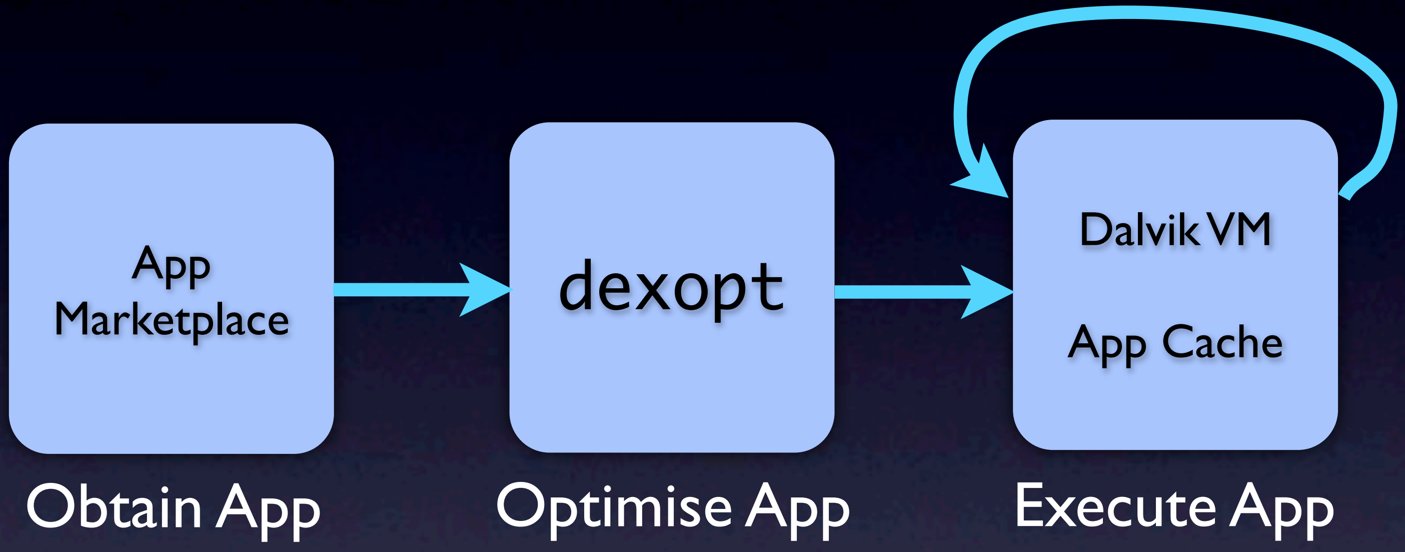
Injectable traces chosen from the hottest JIT traces

Subject to more powerful ahead-of-time optimisations

Installing an App

Stephen Kyle

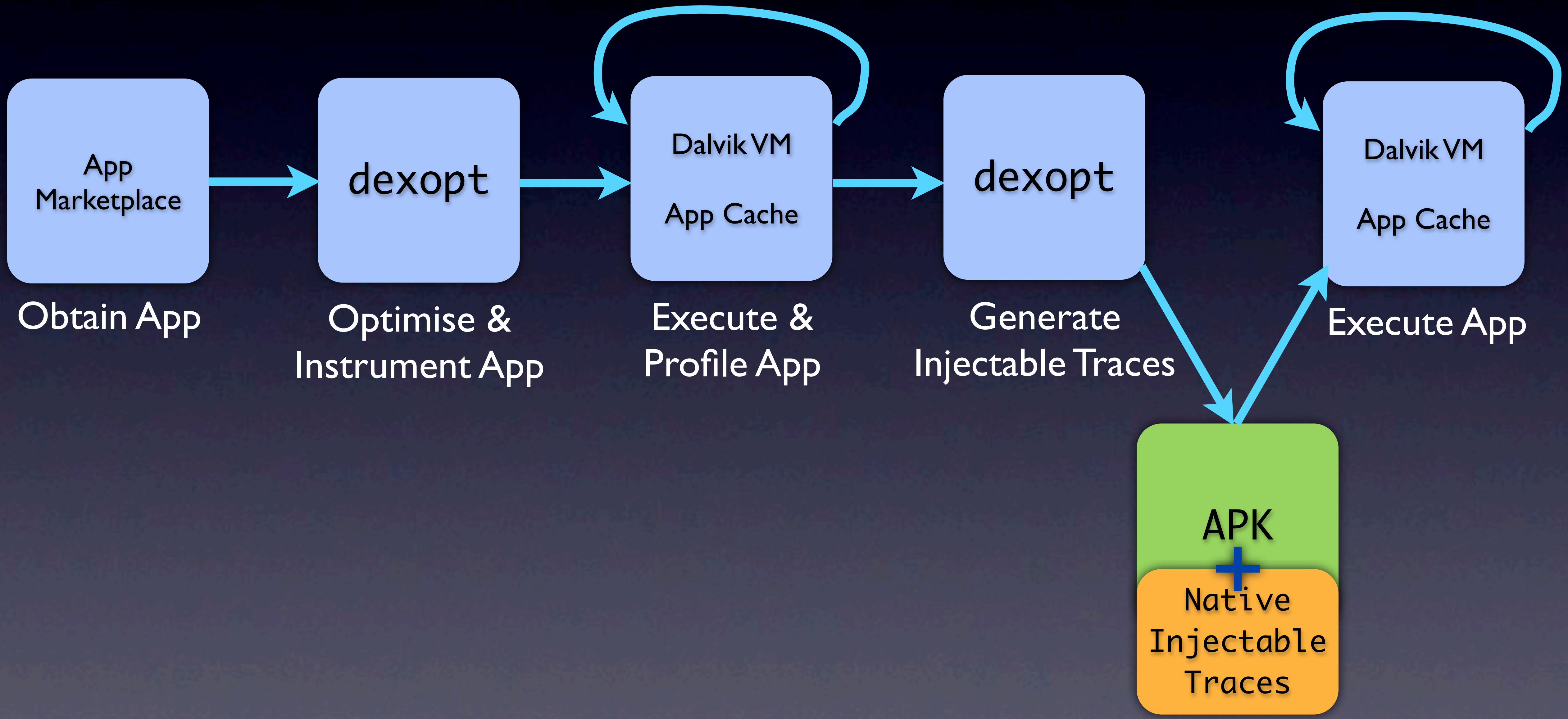
Previously



Installing an App

Stephen Kyle

With Trace Injection



Obtain App

Optimise & Instrument App

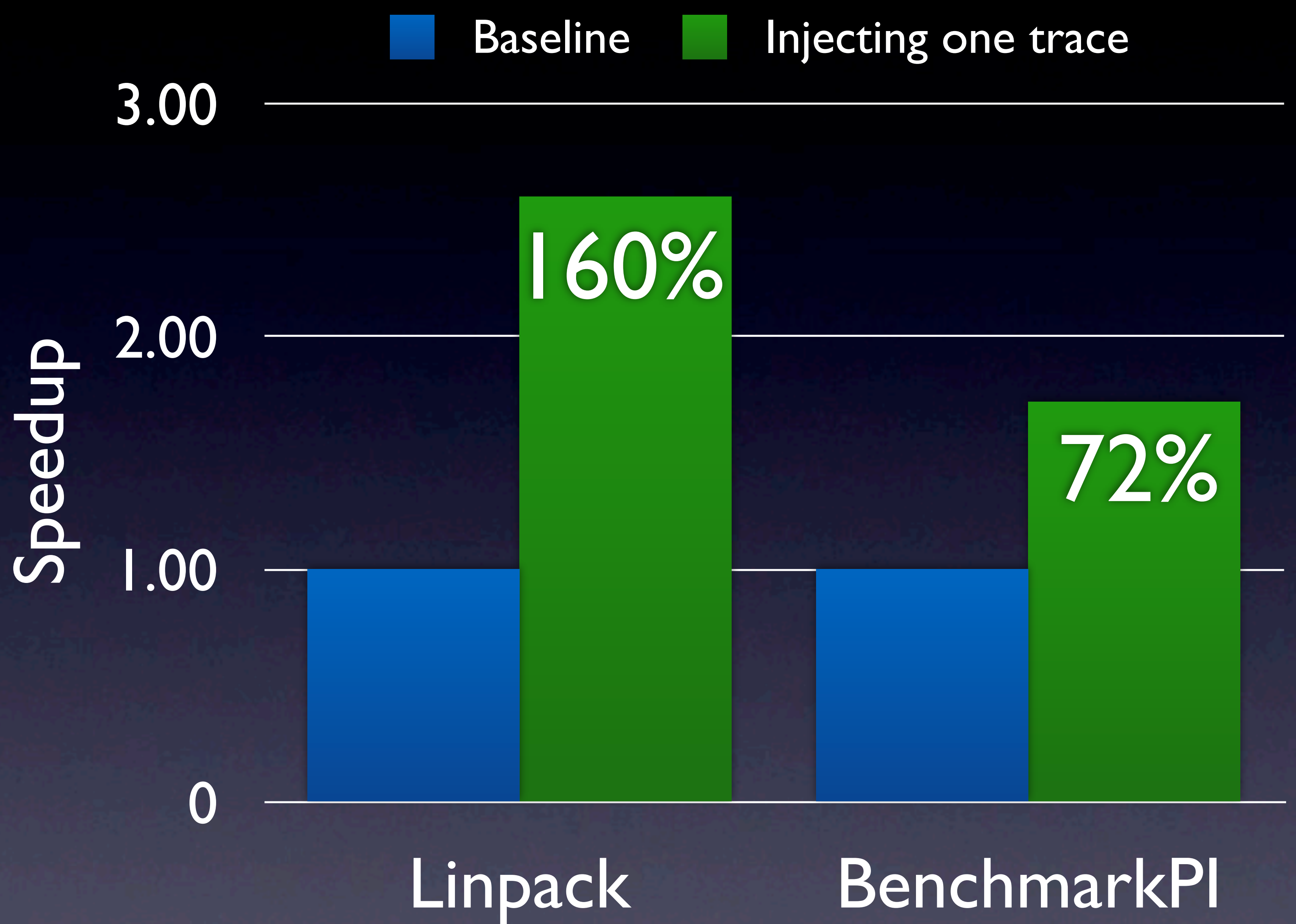
Execute & Profile App

Generate Injectable Traces

Execute App

Preliminary Results

Stephen Kyle



Injecting: Loop Trace - 10 Dalvik Instructions | Linear Trace - 14 Dalvik Instructions

Energy Efficient Scheduling for ASISA Processors

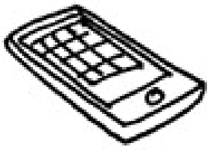
Motivation for Mobile Workload Collection

- Current scheduler decisions are based on a rather narrow field of information

Information about the usage context can help in making more energy efficient scheduling decisions.

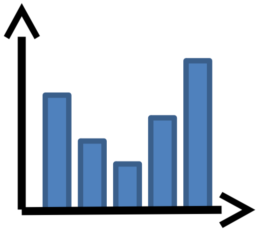
➔ The collected workload must contain both user- and kernel space information to identify the context

Methodology



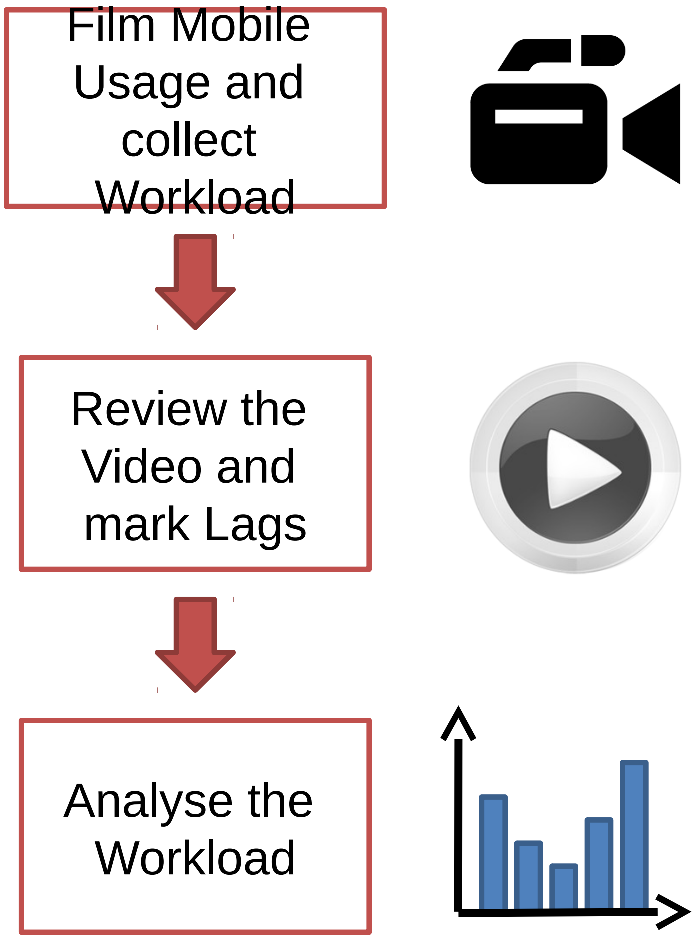
Collect Mobile Workload

Analyse Workload



When does the scheduler get it wrong?

Workload Lag Experiment



Film Mobile Usage and collect Workload

Review the Video and mark Lags

Analyse the Workload

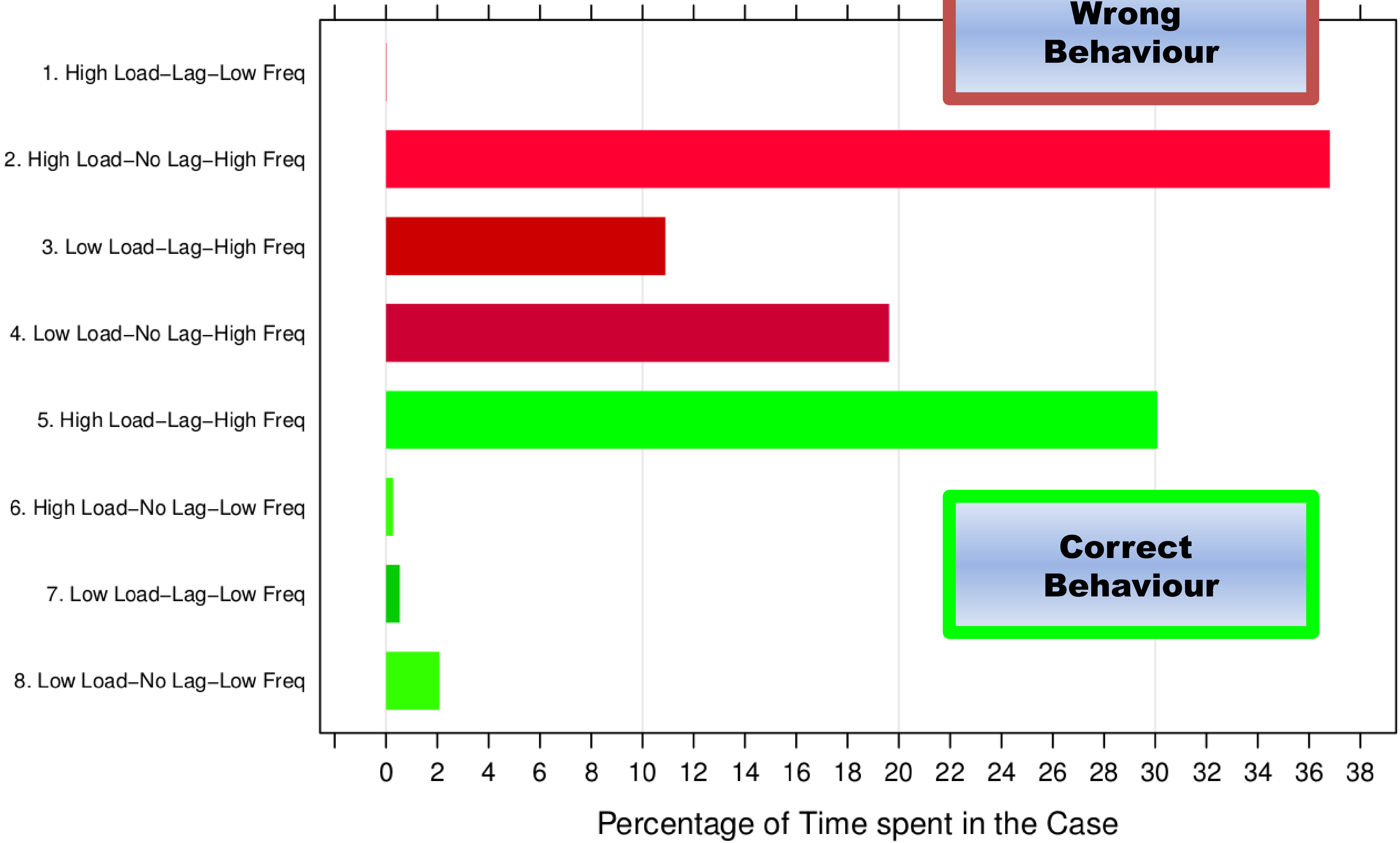
What did the Phone do during a lag?

Was the lag caused by the CPU?
What was the frequency?
What was the load?

Does the Frequency Governor/Scheduler waste energy during those lags?

Energy Efficient Scheduling for ASISA Processors

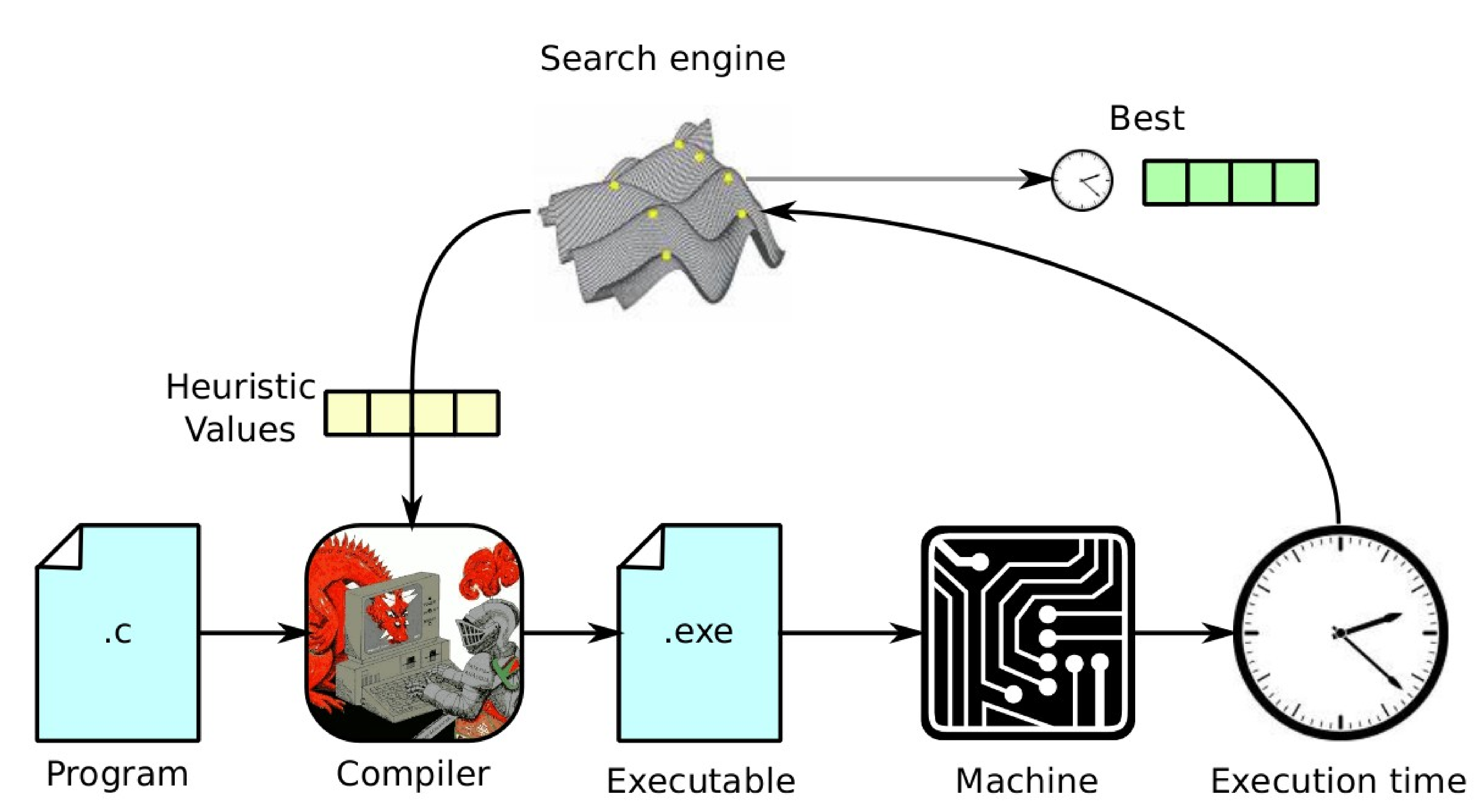
Workload Lag Cases



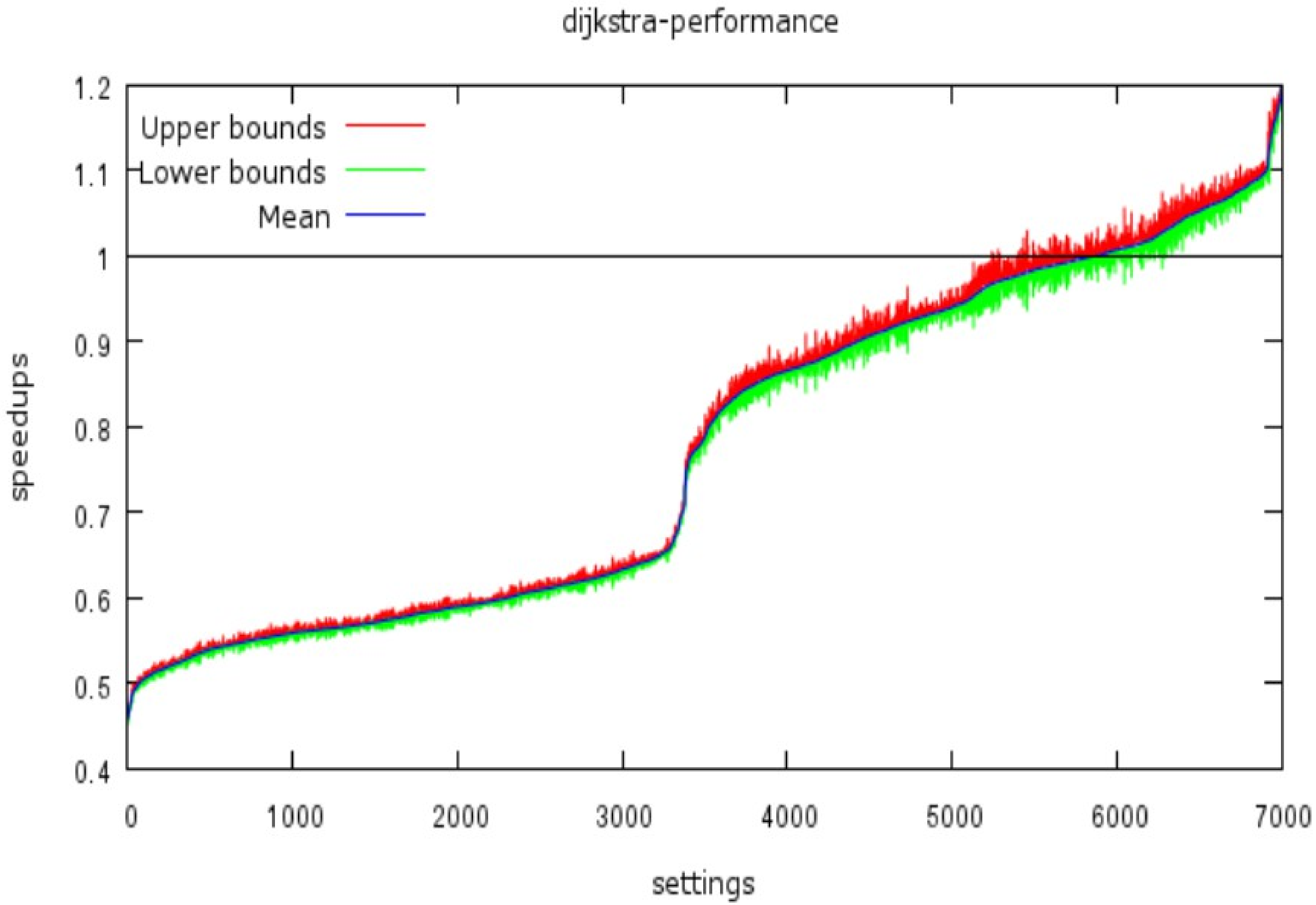
Wrong Behaviour

Correct Behaviour

Iterative Compilation of Native Code



Performance - Dijkstra



Performance

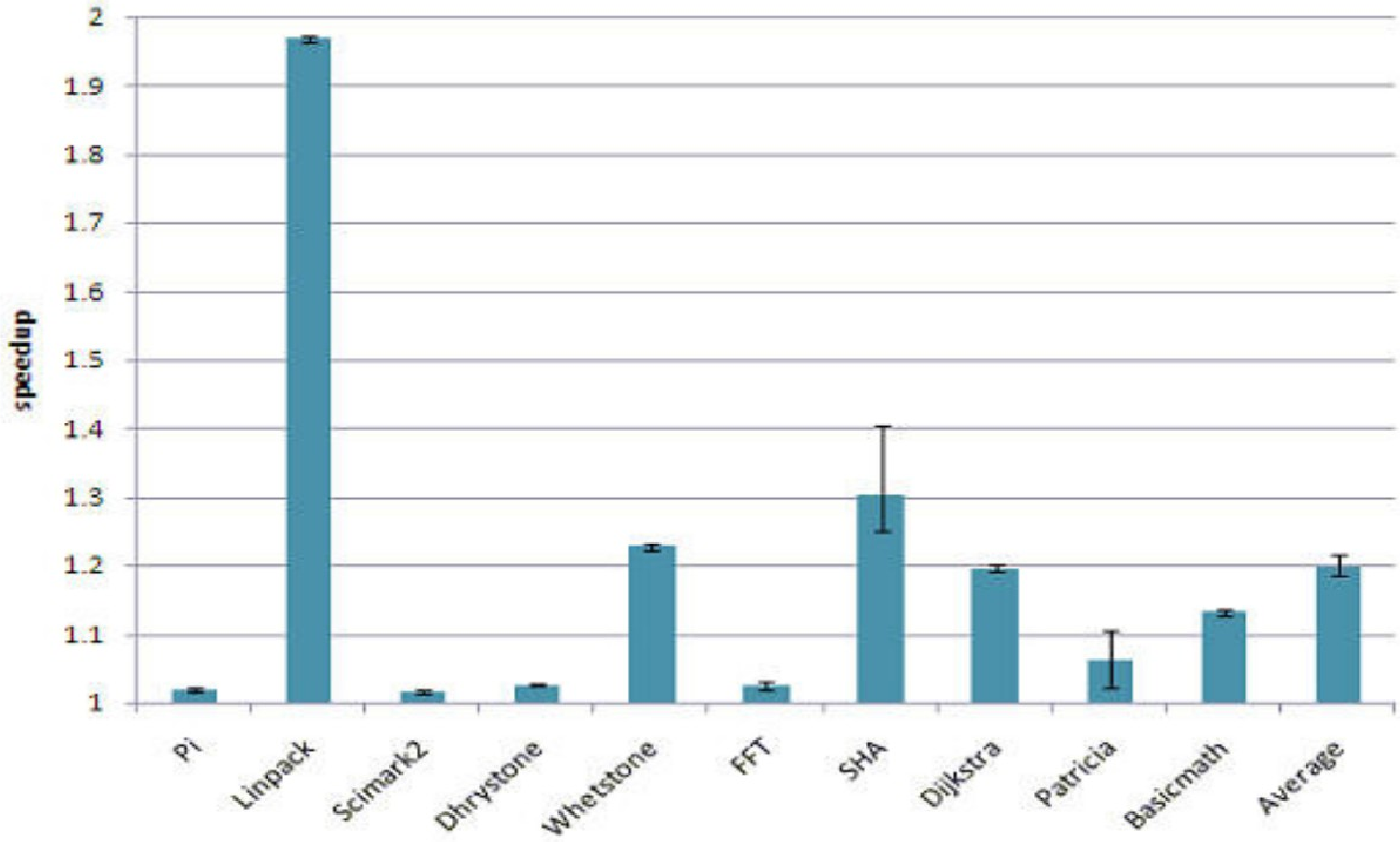
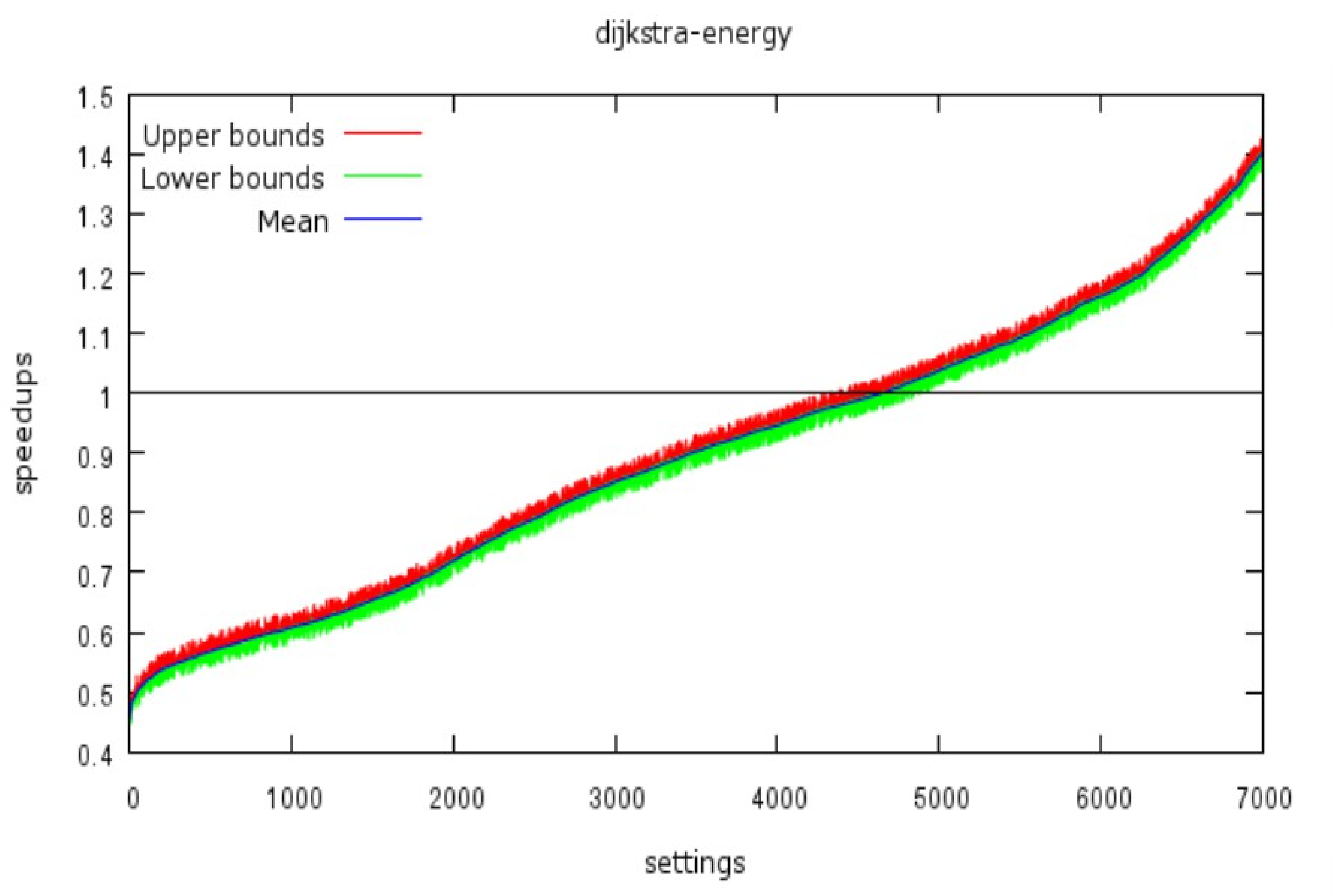


Figure 5.1: Maximum speedups of runtime

Energy - Dijkstra



Energy

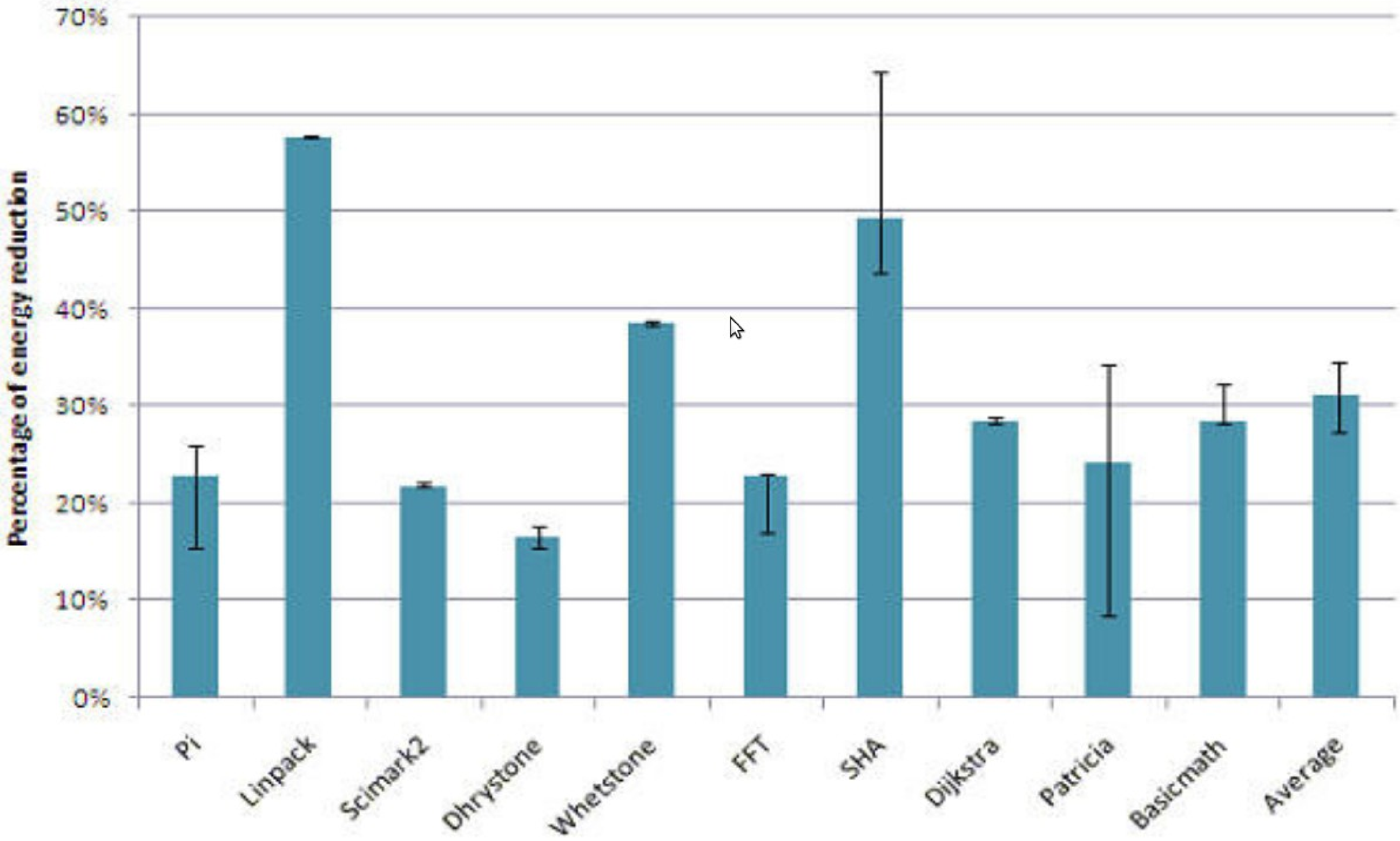
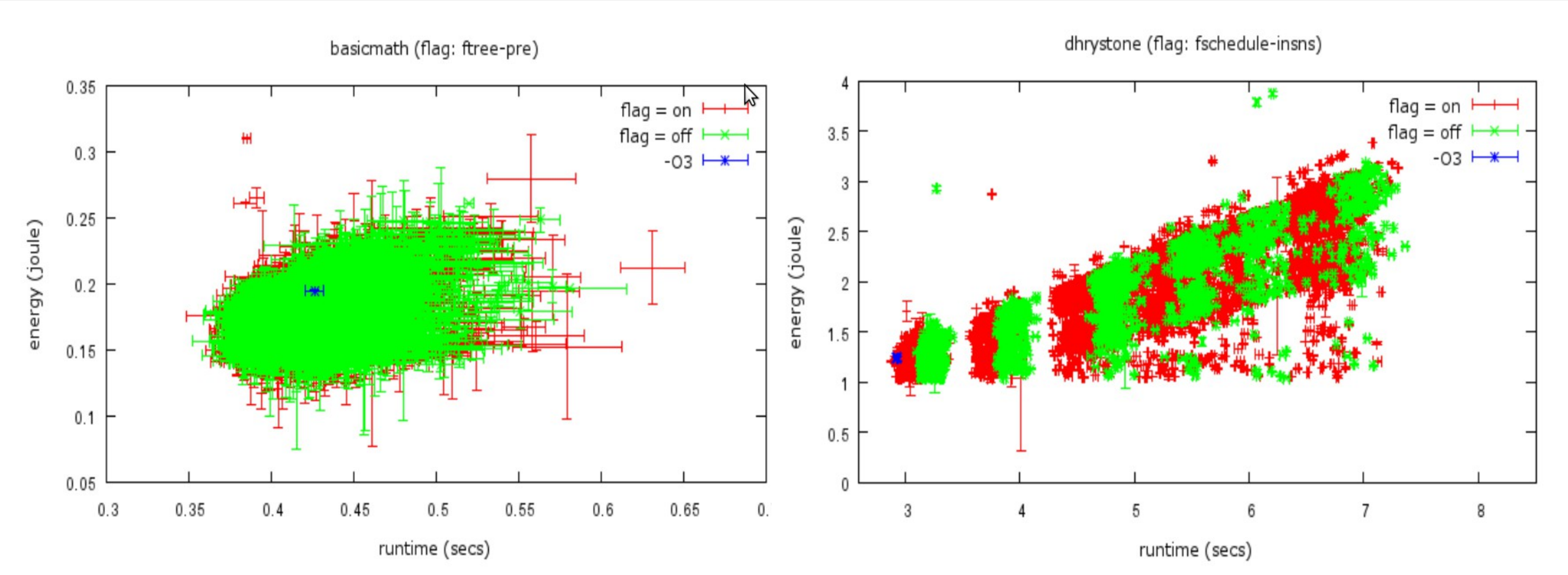


Figure 5.2: Maximum energy improvement rates

Energy vs Performance

- Are energy and performance correlated?



- Not really! Why?
- If could predict recharge time, change version