

Architecture-Parametric Timing Analysis

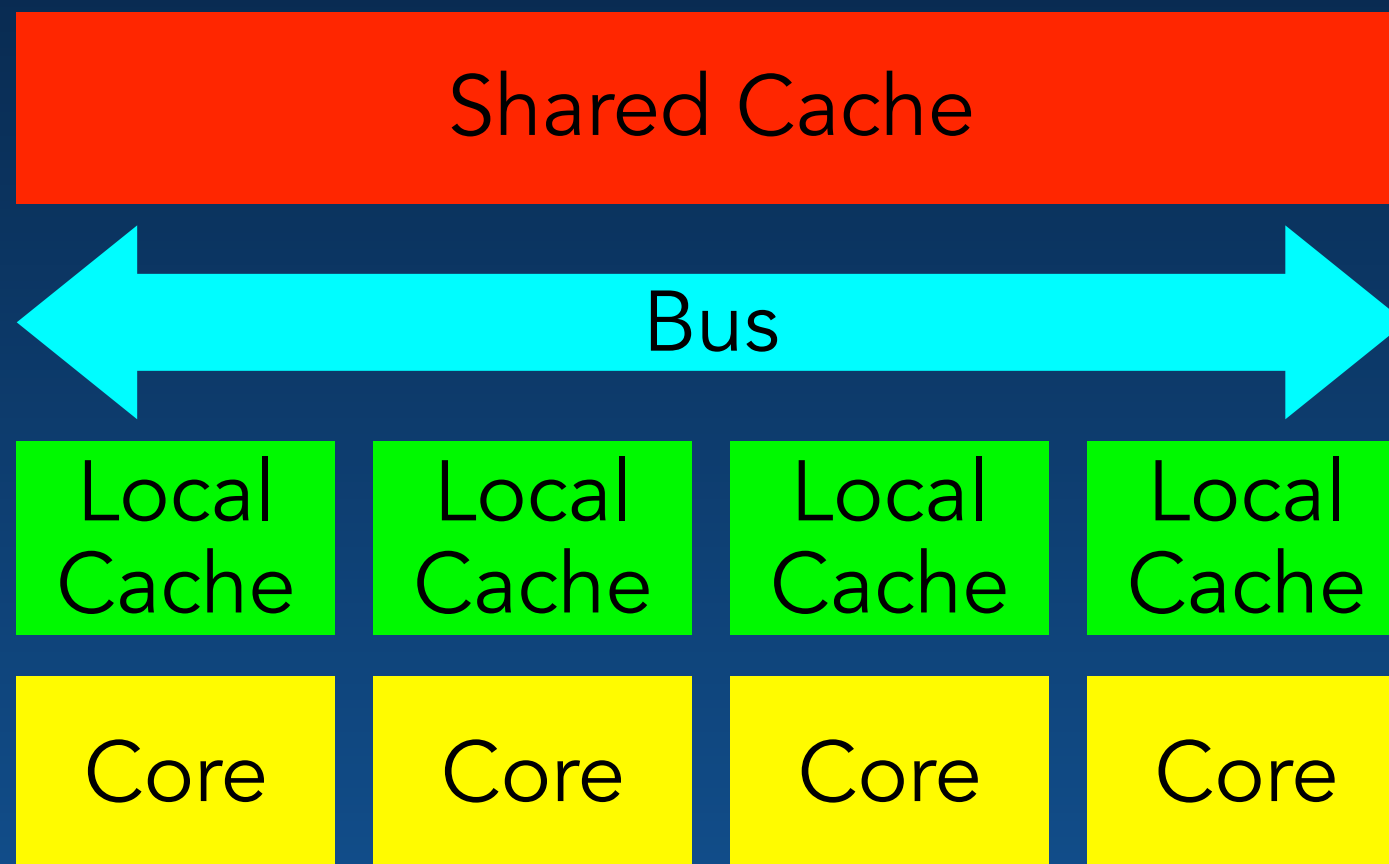
Jan Reineke

Johannes Doerfert

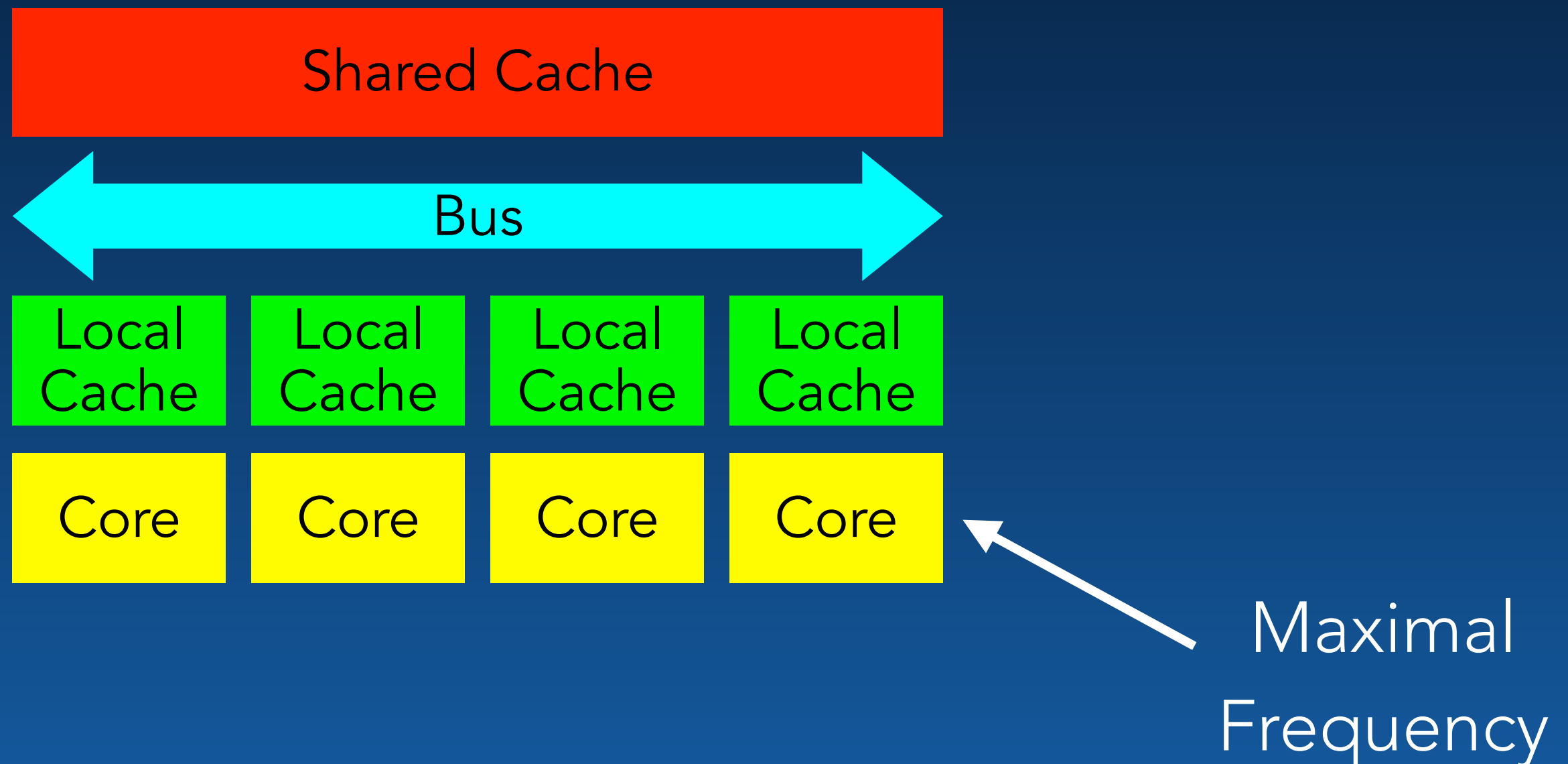


20th IEEE Real-Time and Embedded Technology and Applications Symposium
April 15-17, 2014
Berlin, Germany

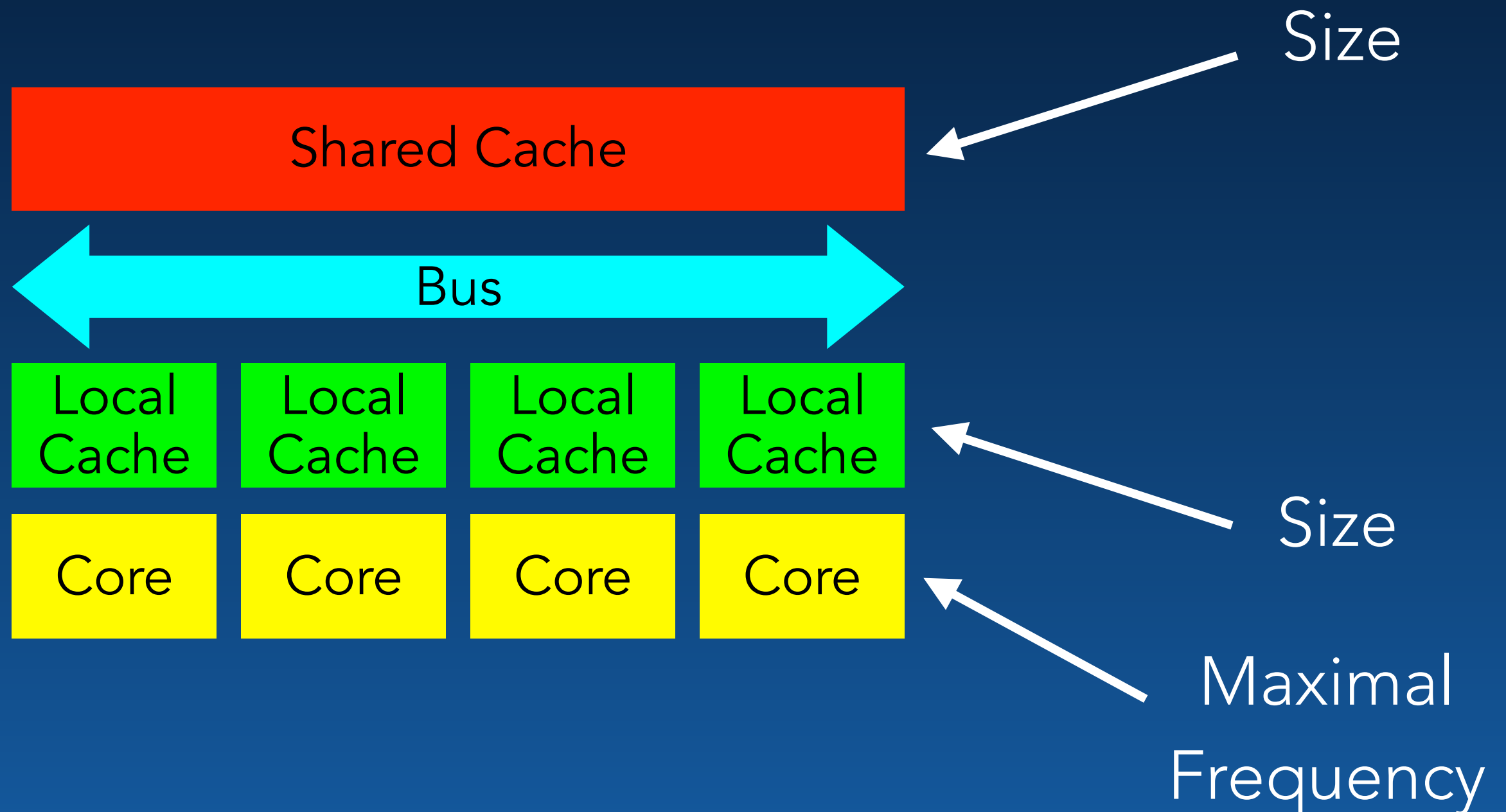
Architecture-Configuration Challenge: At Design Time



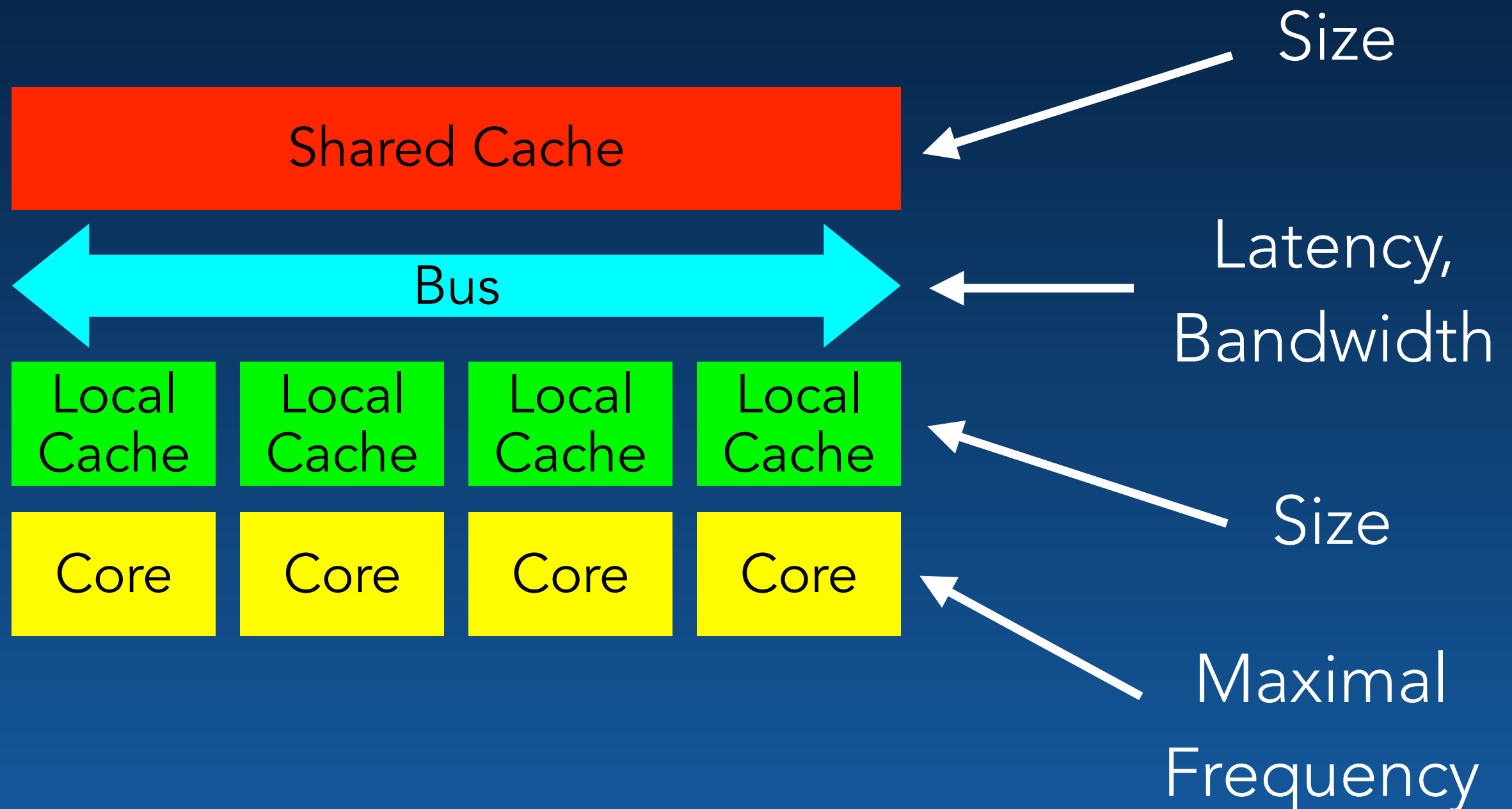
Architecture-Configuration Challenge: At Design Time



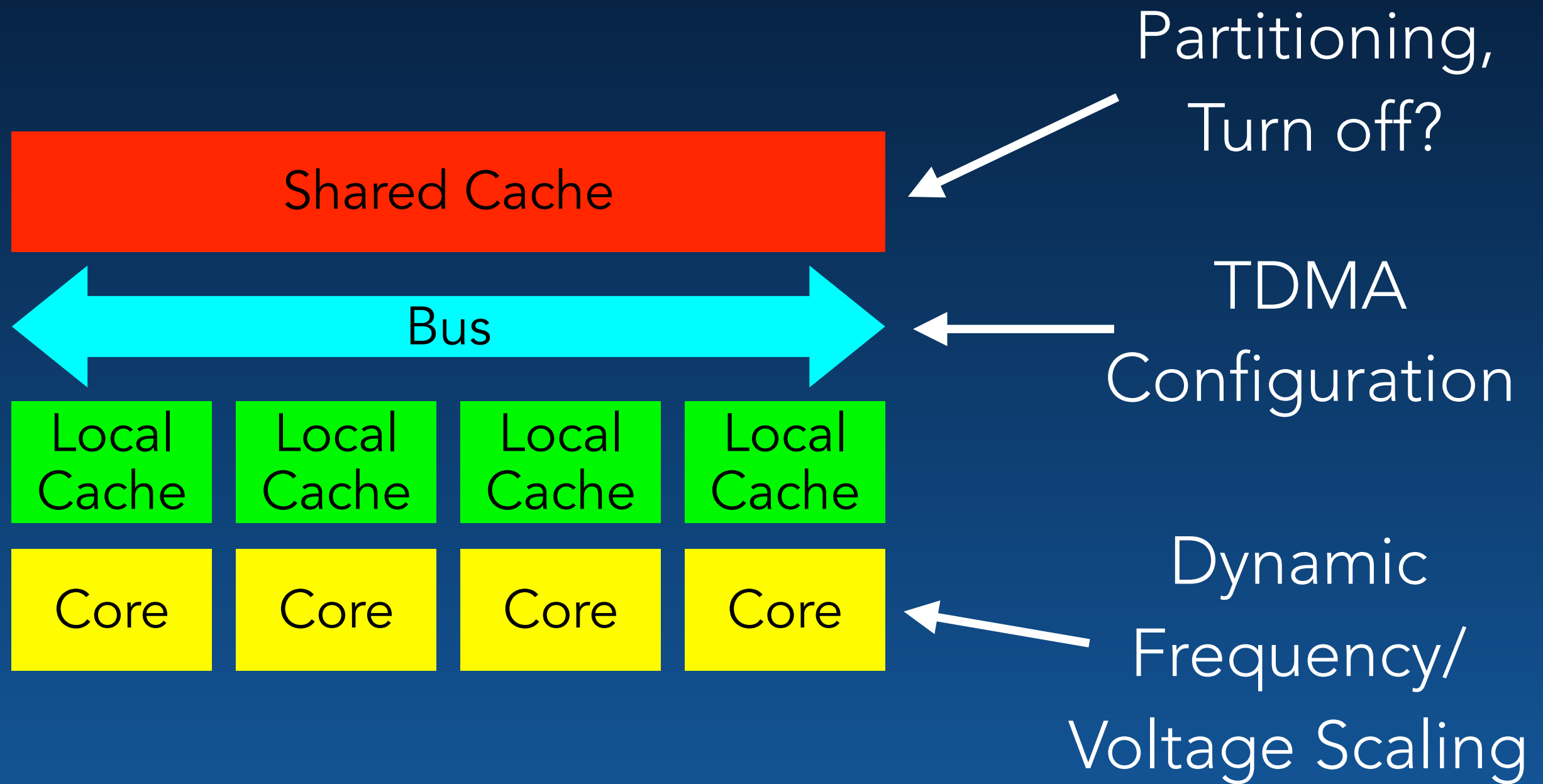
Architecture-Configuration Challenge: At Design Time



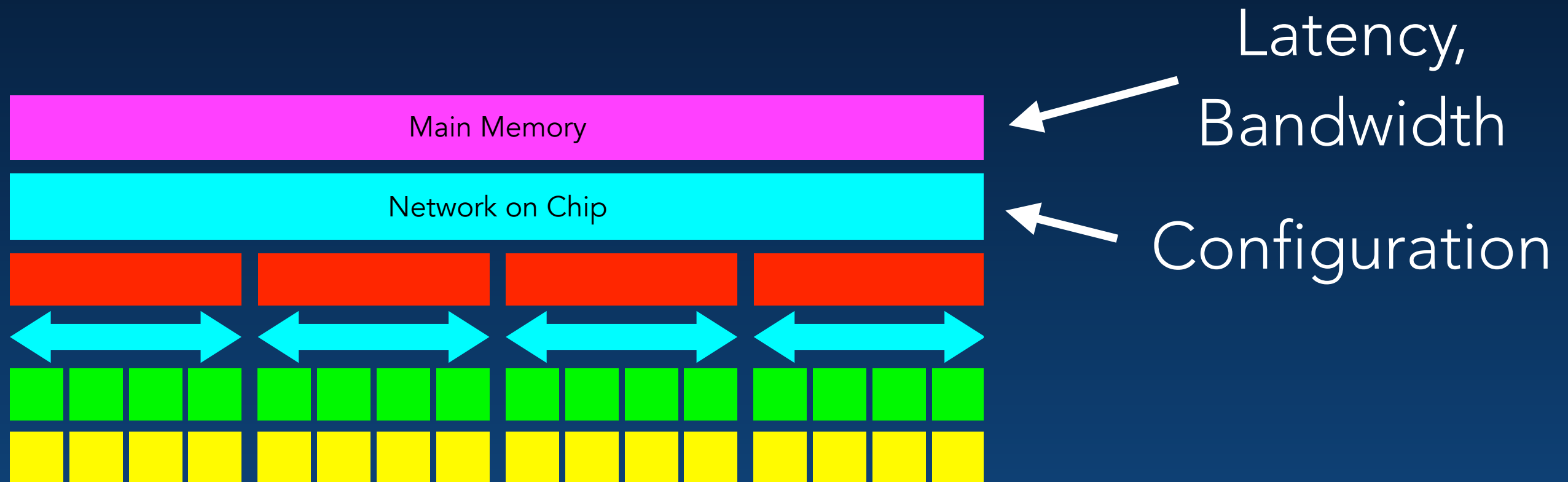
Architecture-Configuration Challenge: At Design Time



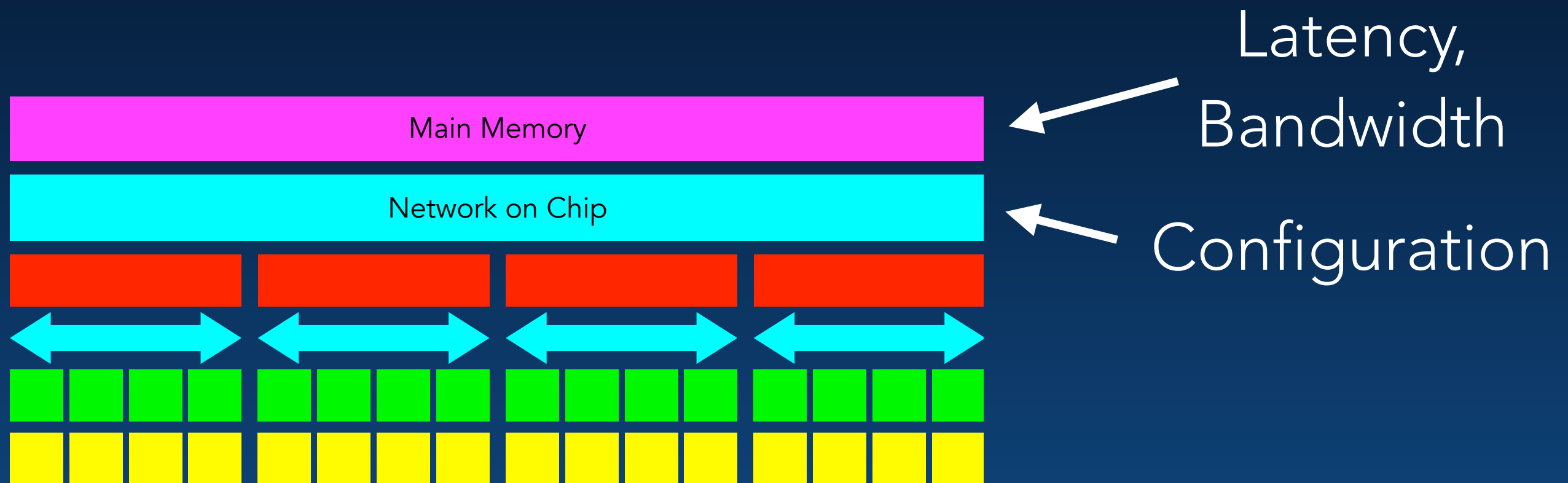
Architecture-Configuration Challenge: At Runtime



Architecture-Configuration Challenge: Many-Core



Architecture-Configuration Challenge: Many-Core



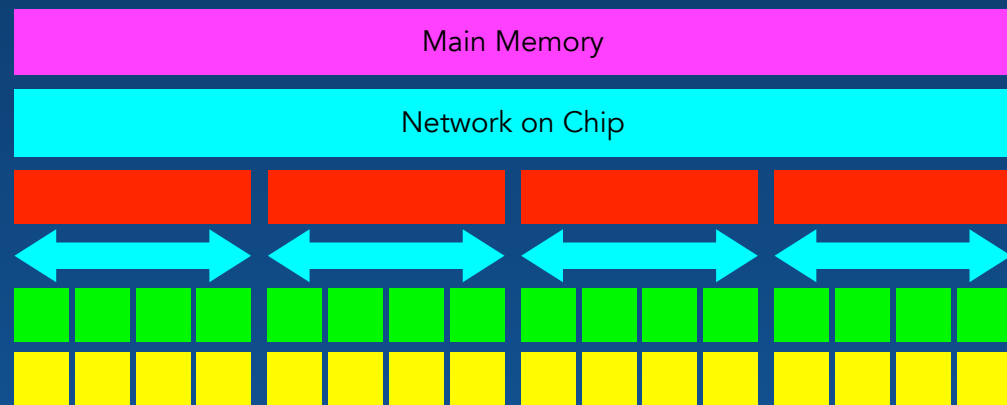
Configuration affects **implementation cost**, **energy consumption**, and **worst-case execution times!**

Architecture-Parametric Timing Analysis

```
// Perform the convolution.  
for (int i=0; i<10; i++) {  
    x[i] = a[i]*b[j-i];  
    // Notify listeners.  
    notify(x[i]);  
}
```

Embedded Software

+



Configurable Platform



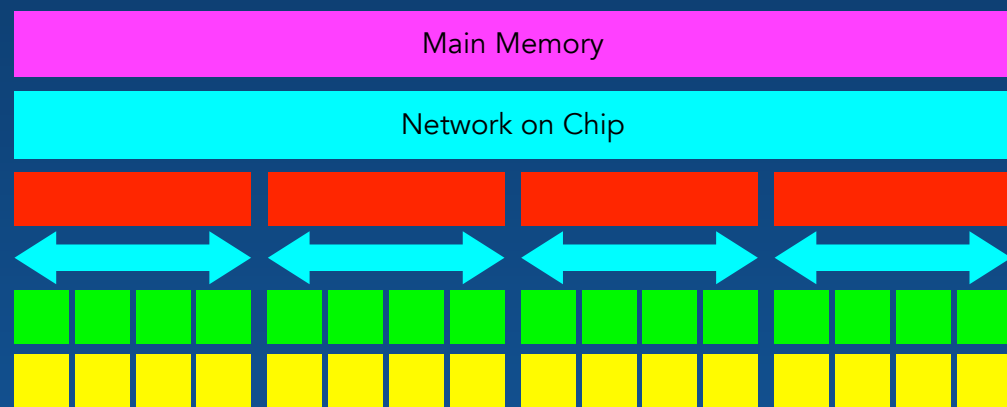
Parametric WCET

Architecture-Parametric Timing Analysis

```
// Perform the convolution.  
for (int i=0; i<10; i++) {  
    x[i] = a[i]*b[j-i];  
    // Notify listeners.  
    notify(x[i]);  
}
```

Embedded Software

+



Configurable Platform

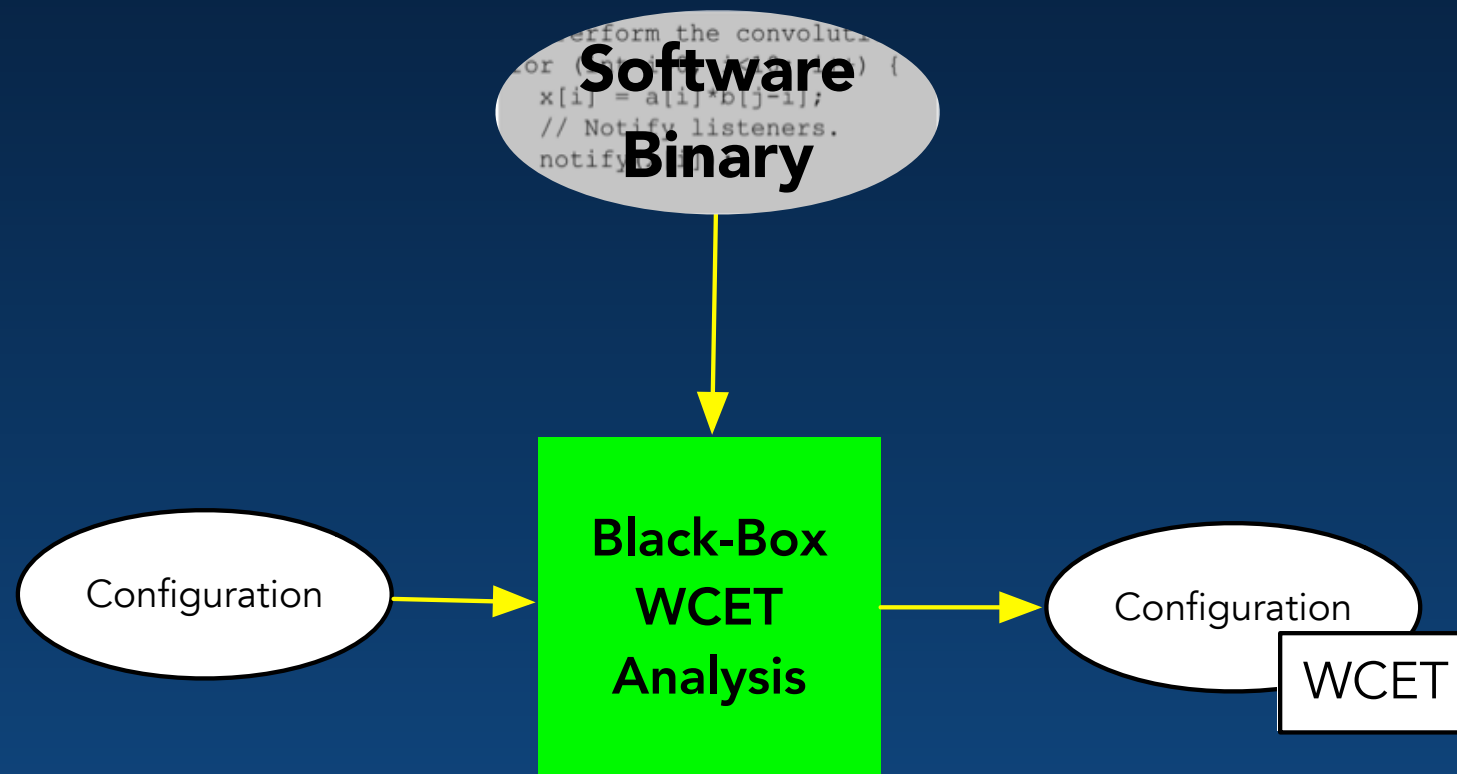


Parametric WCET

Desiderata:

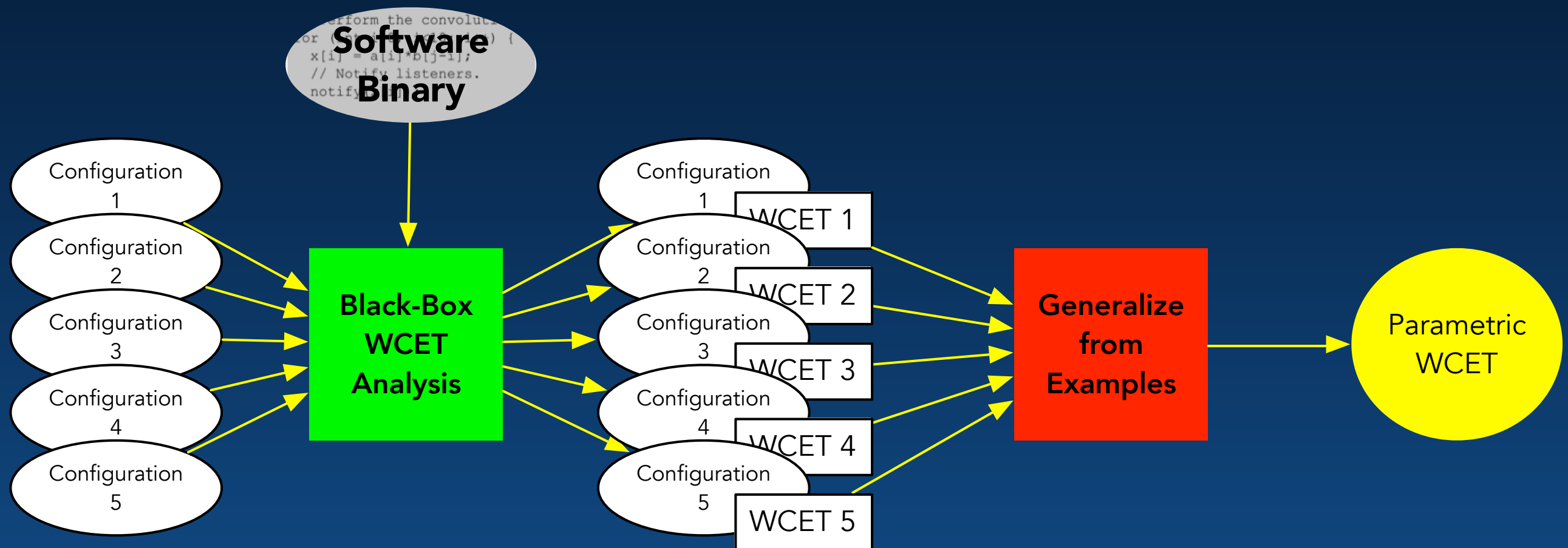
- Precise
- Efficiently evaluable

Architecture-Parametric Timing Analysis: "Black-Box" Approach

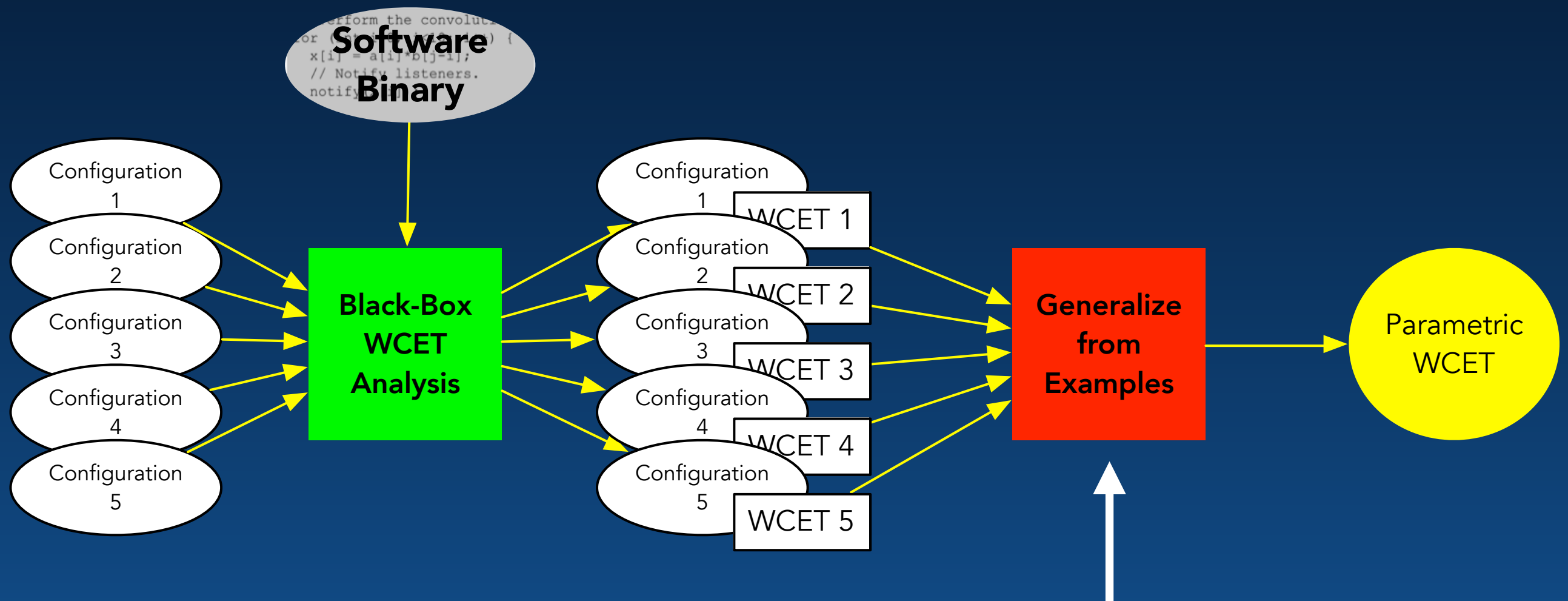


Conventional
non-parametric
timing analysis

Architecture-Parametric Timing Analysis: "Black-Box" Approach

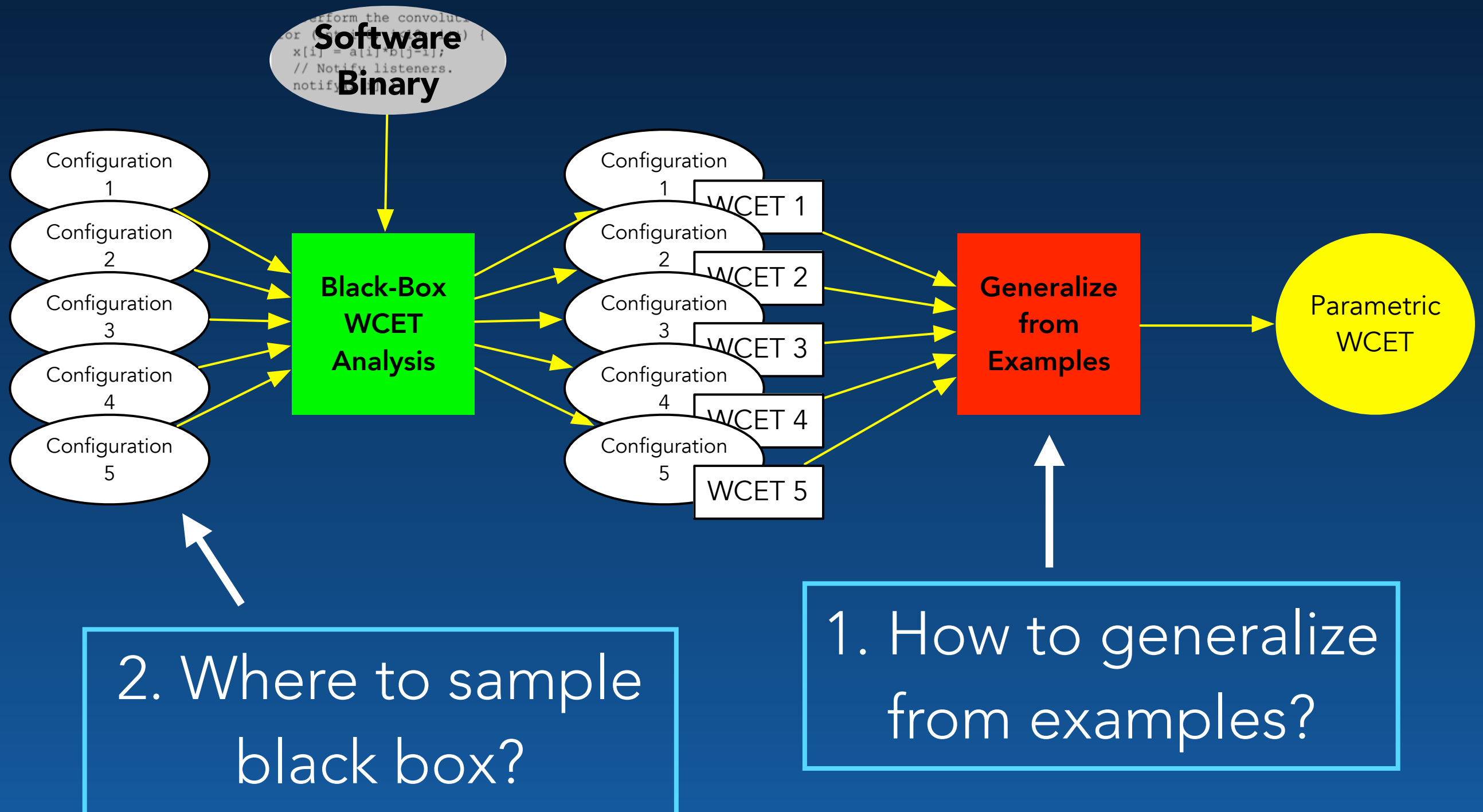


Architecture-Parametric Timing Analysis: "Black-Box" Approach



1. How to generalize from examples?

Architecture-Parametric Timing Analysis: "Black-Box" Approach



Requirements for Sound and Efficient Generalization

Necessary:

Execution times should be **monotone** in parameters:

“higher frequencies yield shorter execution times”

“smaller caches yield longer execution times”

Requirements for Sound and Efficient Generalization

Desirable for efficiency:

Execution time should depend **linearly** on parameters:

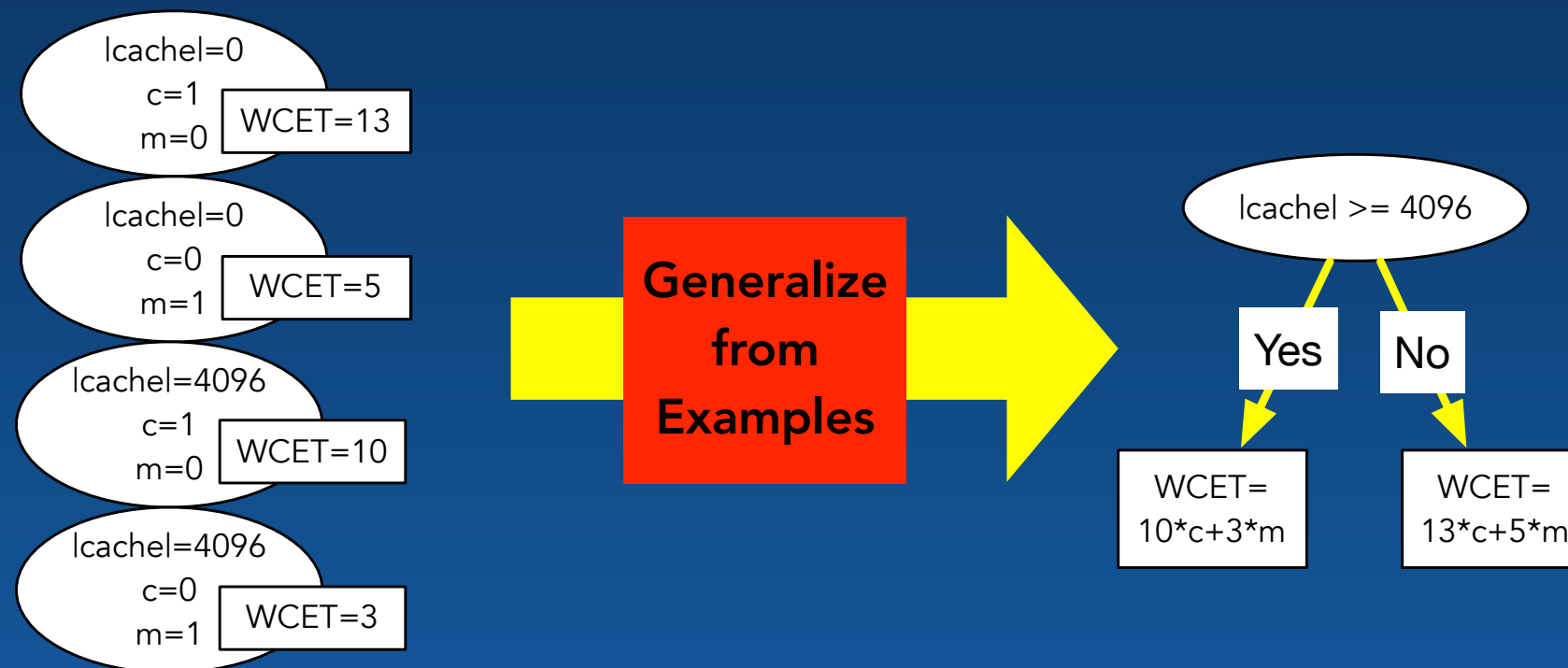
“doubling the processor frequency will decrease execution time by a factor of two”

1. How to Generalize from Examples?

Reduction to Parametric Linear Programming

Formulate a series of **parametric linear programs**, encoding:

- Configurations/WCETs obtained from Black Box
- Properties that allow to generalize

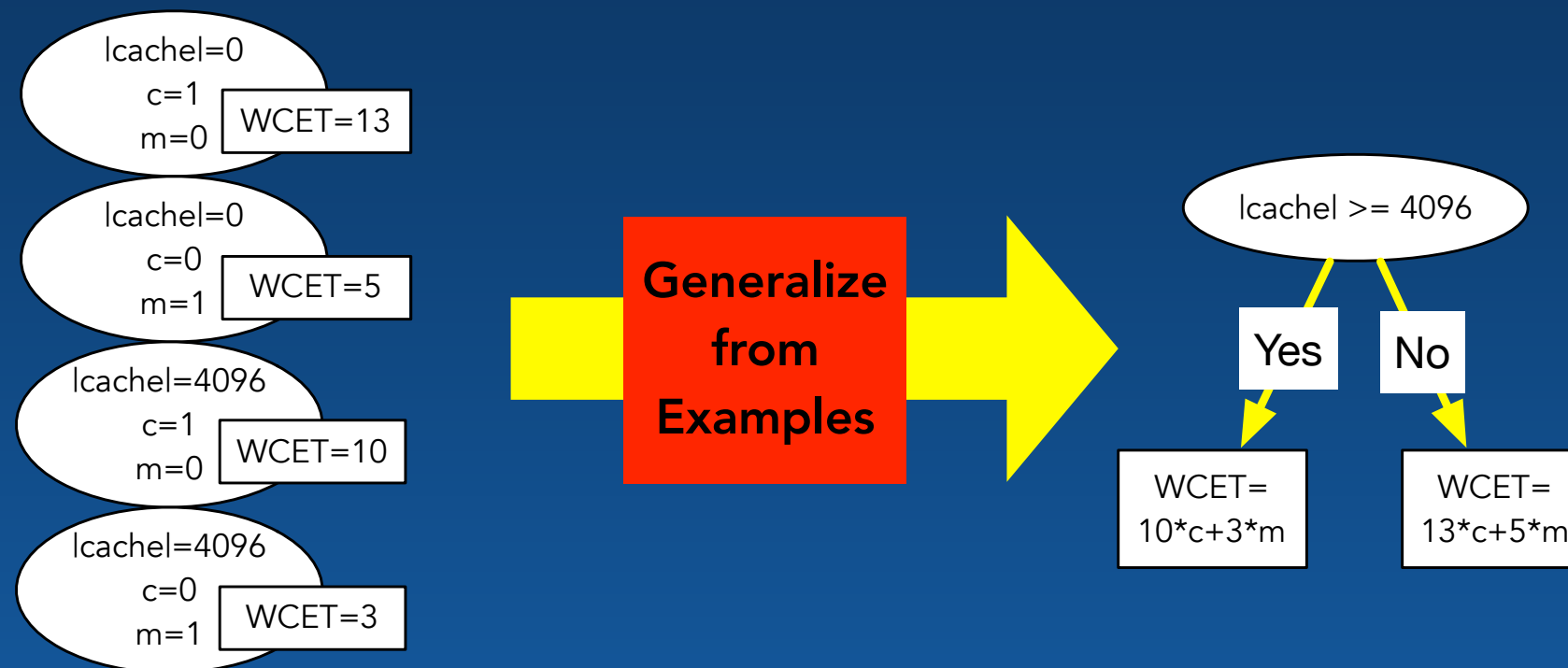


1. How to Generalize from Examples?

Reduction to Parametric Linear Programming

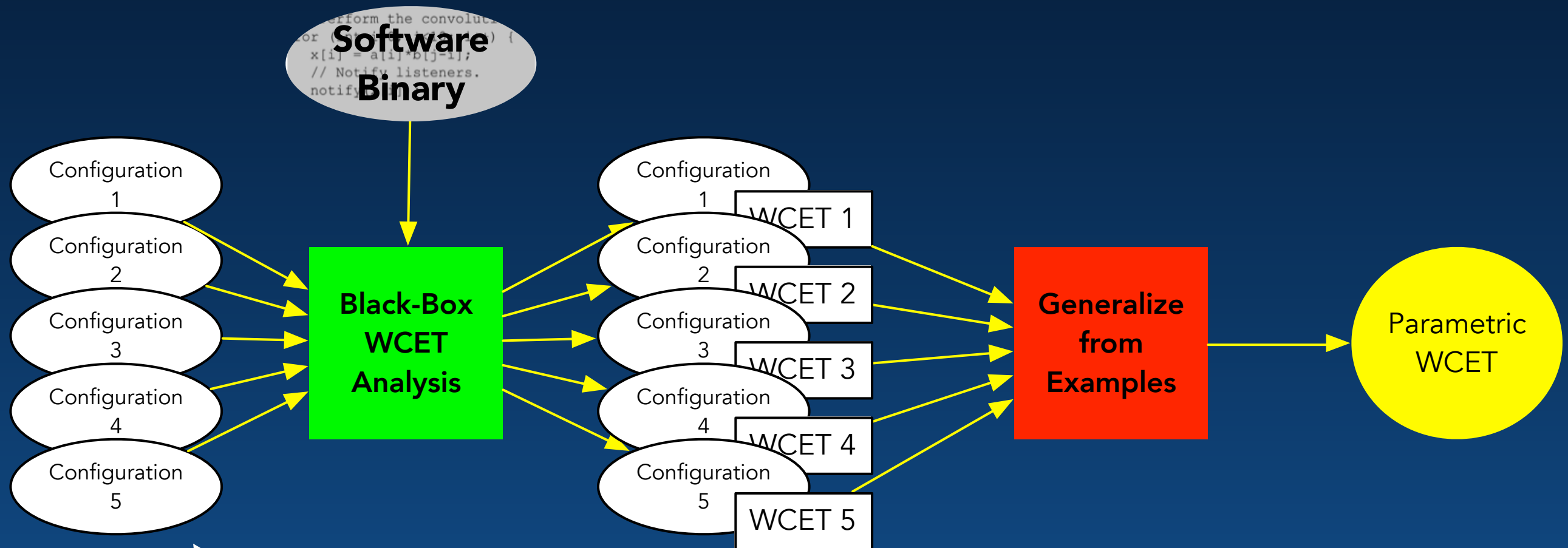
Formulate a series of **parametric linear programs**, encoding:

- Configurations/WCETs obtained from Black Box
- Properties that allow to generalize



See paper for details!

"Black-Box" Approach



2. Where to sample
black box?

2. Where to sample the black box?

Wanted:

Small set of configurations that yields



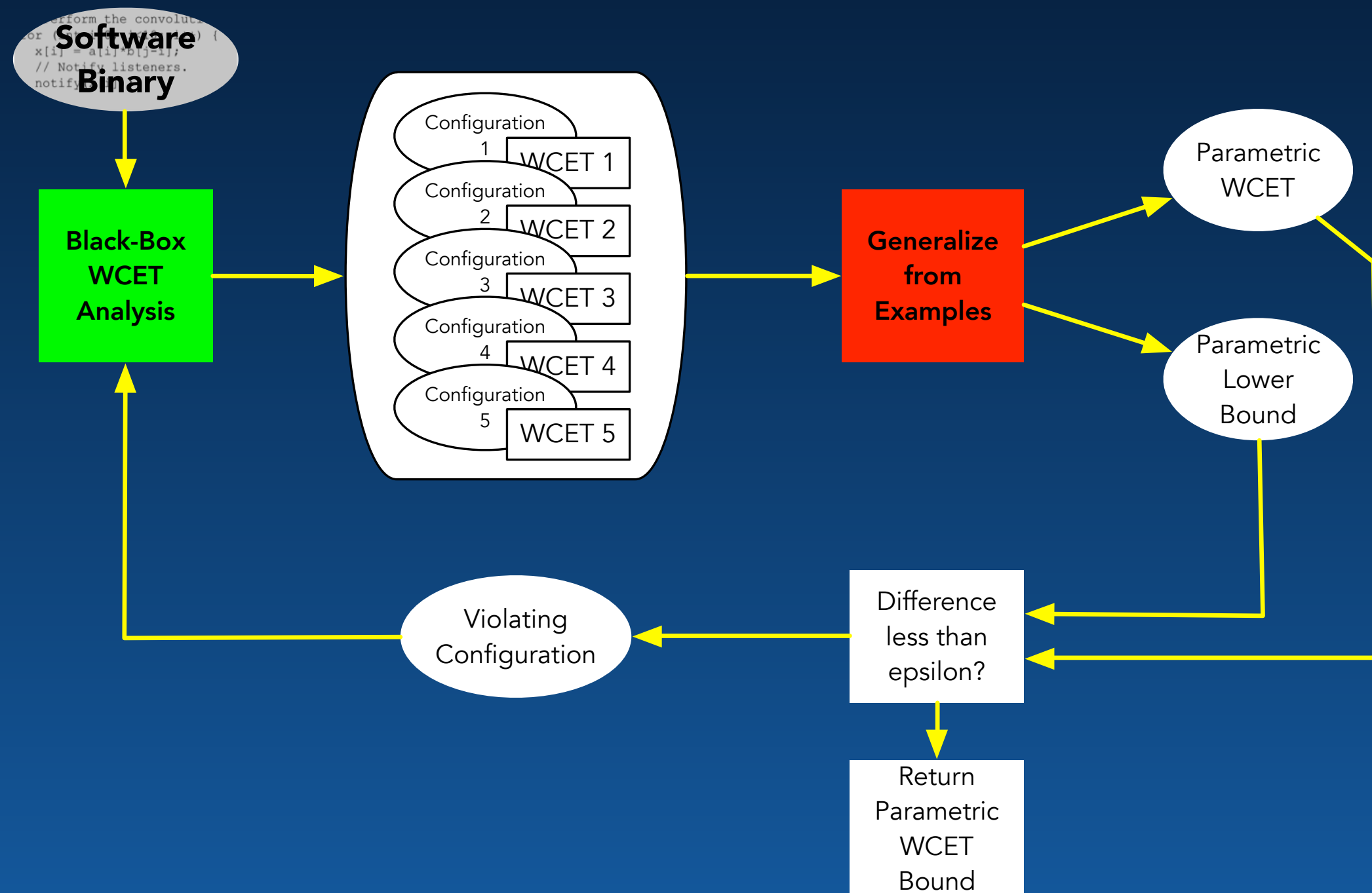
fast
analysis

precise parametric WCET

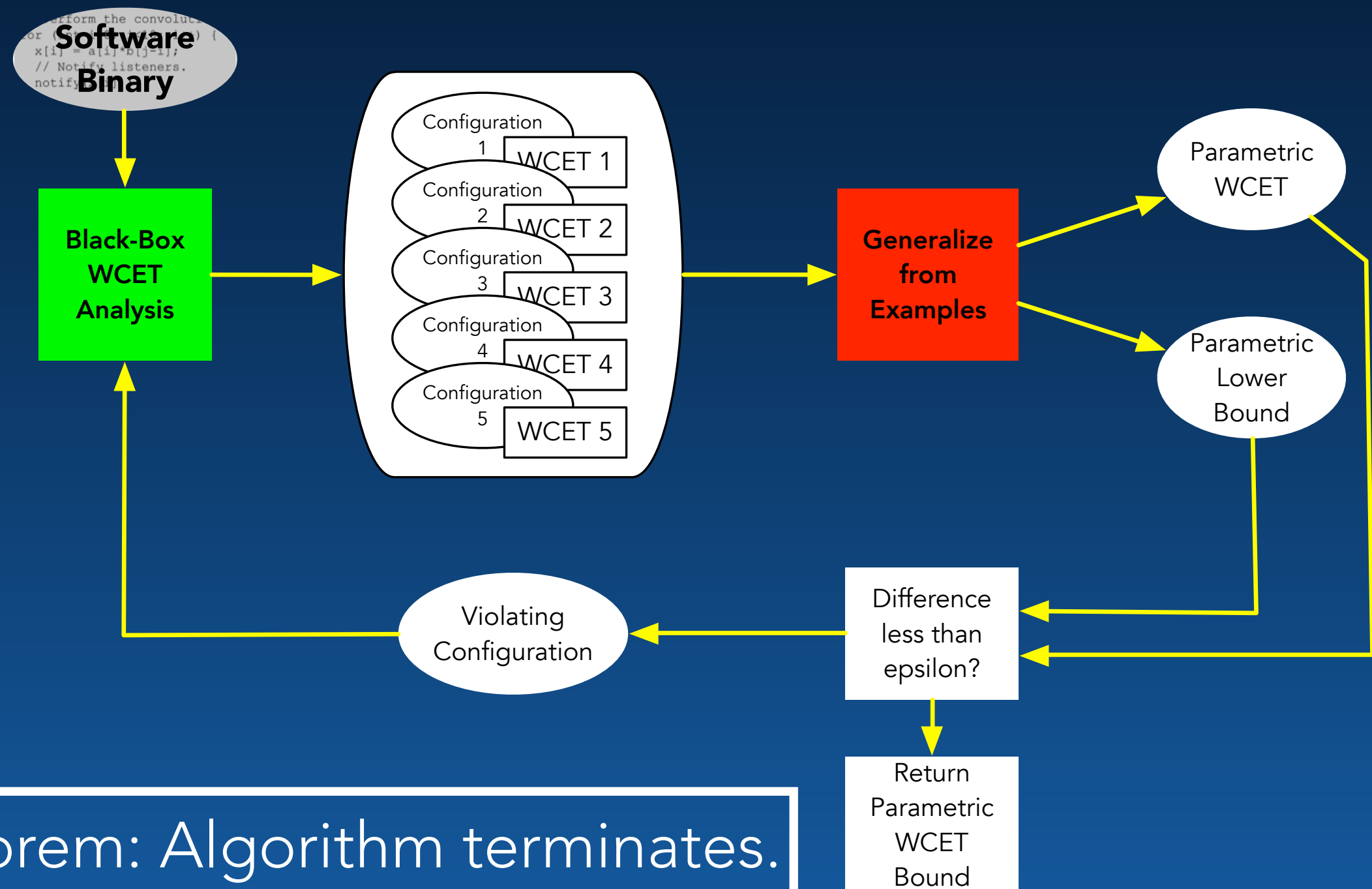
=

"close" to
black-box
"everywhere"

2. Where to sample the black box? Incremental Sampling



2. Where to sample the black box? Incremental Sampling



Theorem: Algorithm terminates.

Target for Prototype: A Parameterized Precision-Timed Architecture

Parameterized version of the PTARM, a **predictable microarchitecture** developed within the PRET project.

6 parameters that control

- latencies of arithmetic and branch instructions,
- latencies of loads and stores to the scratchpads and to DRAM,
- sizes of instruction and data scratchpads.

Target for Prototype: A Parameterized Precision-Timed Architecture

Parameterized version of the PTARM, a **predictable microarchitecture** developed within the PRET project.

6 parameters that control

- latencies of arithmetic and branch instructions,
- latencies of loads and stores to the scratchpads and to DRAM,
- sizes of instruction and data scratchpads.

For experimental evaluation:

- Black-box WCET analysis based on OTAWA
- Parameterized PTARM simulator

Experimental Evaluation: Precision of Black Box

Name	Black Box (cycles)	Simulator (cycles)	Ratio
adpcm	9989637	1598152	6.25
bs	318	279	1.14
bsort100	998109	8293	120.36
crc	248231	116995	2.12
fdct	11262	11069	1.02
fibcall	1140	1131	1.01
insertsort	4965	2949	1.68
janne_complex	4048	753	5.38
jfdctint	14016	13951	1.00
matmult	755274	745669	1.01
ns	42550	42549	1.00
nsichneu	32339	15551	2.08
qsort-exam	2132100	11125	191.65
statemate	108766	2809	38.72

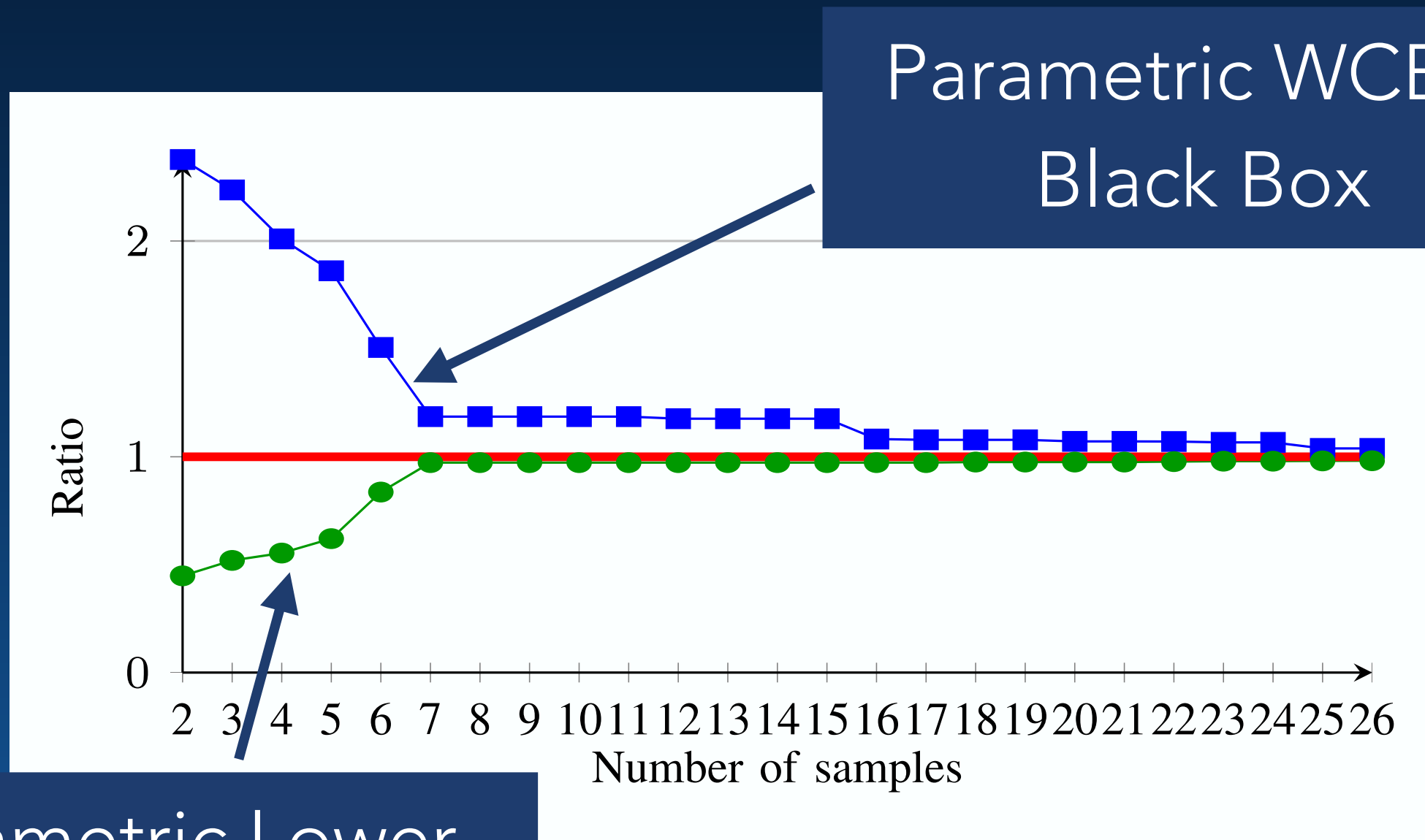


Mälardalen benchmarks
minus floating-point,
recursion, complex
switch statements



Black Box/Simulator

Experimental Evaluation: Precision in Terms of Number of Samples

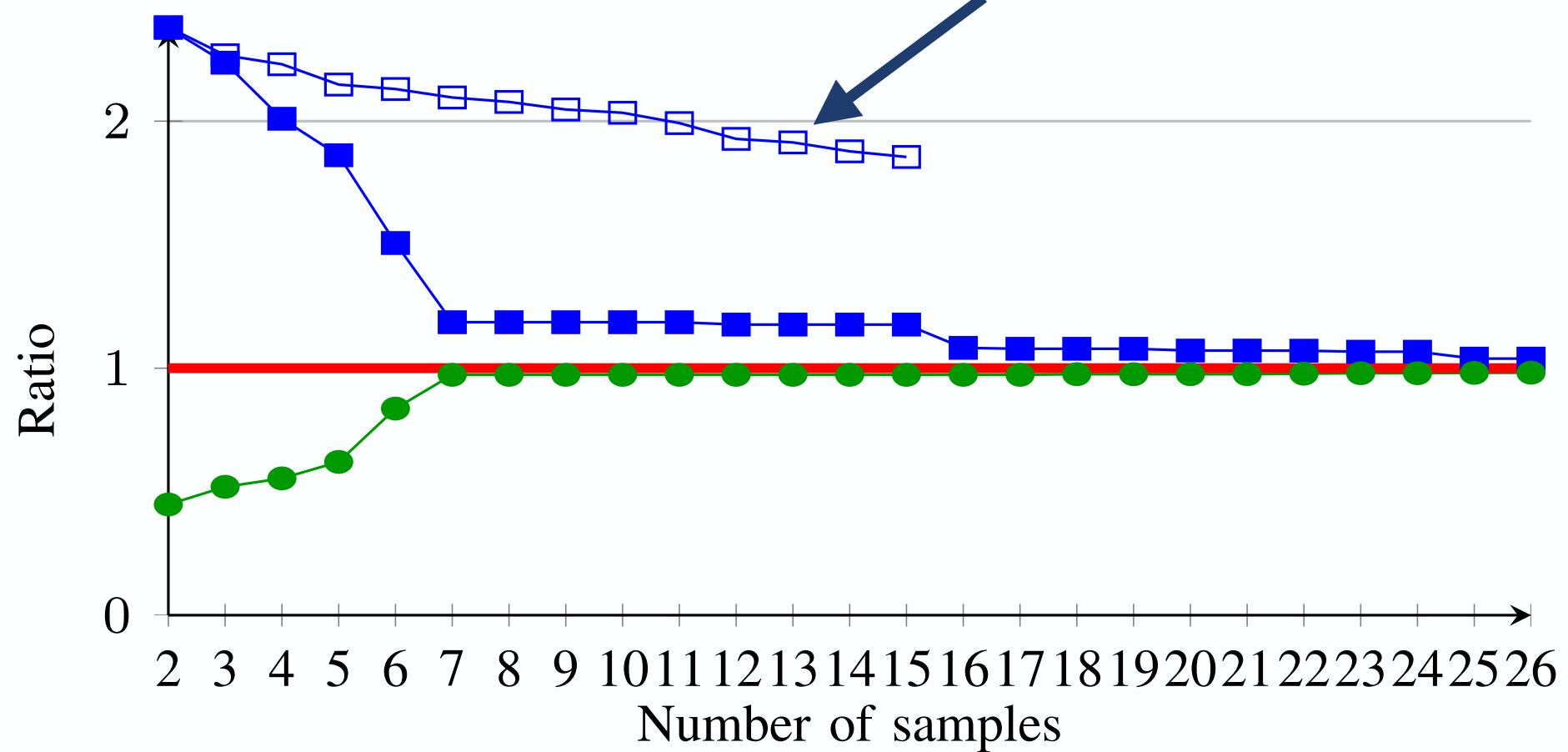


Parametric Lower
Bound/Black Box

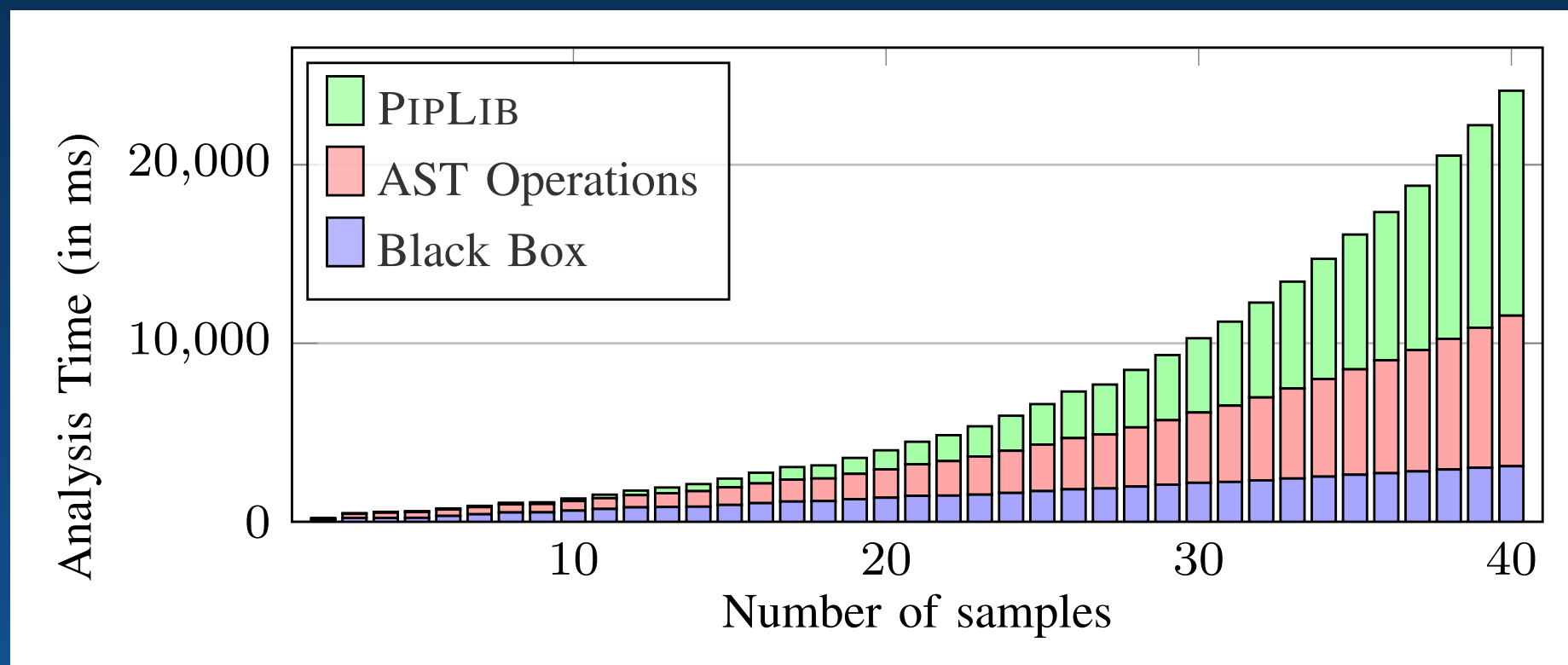
Geometric Mean

Experimental Evaluation: Versus Random Sampling

Random Samples

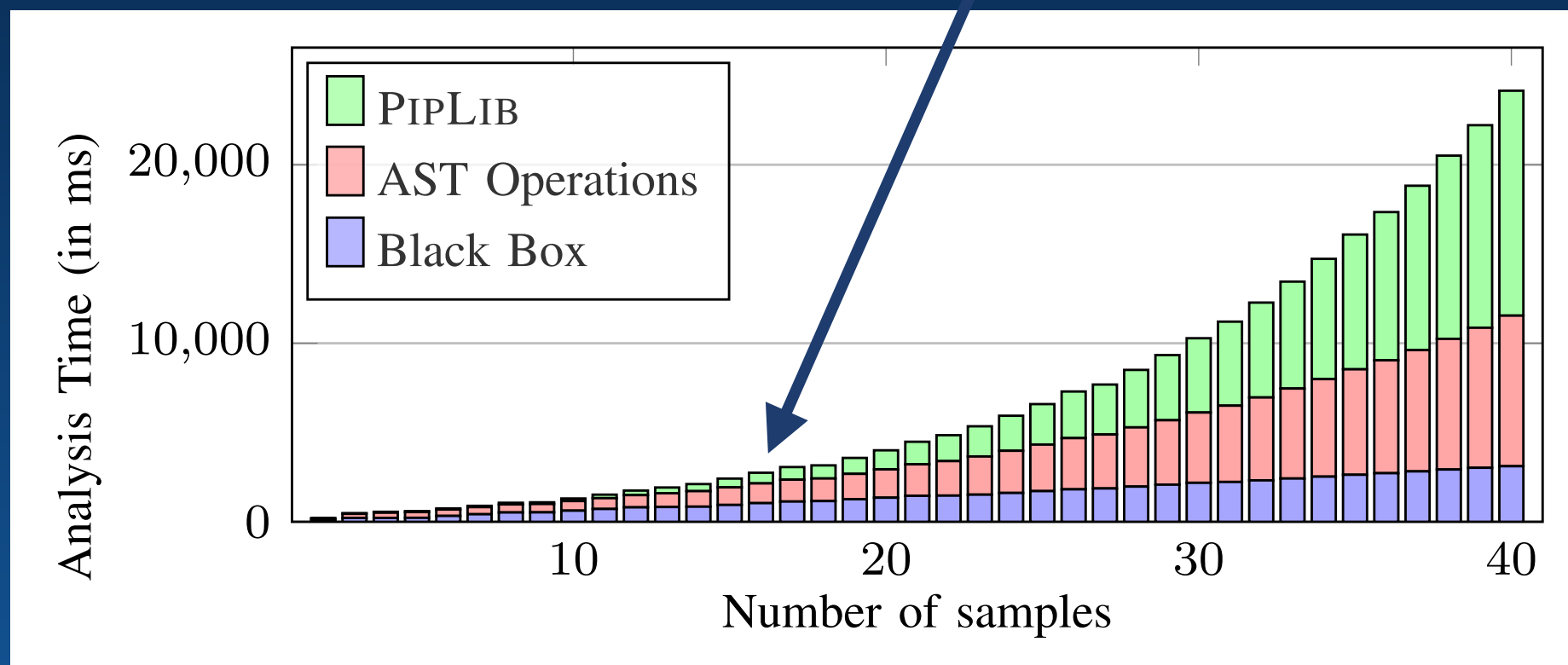


Experimental Evaluation: Analysis Time in Terms of Number of Samples



Experimental Evaluation: Analysis Time in Terms of Number of Samples

16 Samples:
~ 2.2 seconds



Conclusions and Future Work

First general framework for
architecture-parametric timing analysis.

Future Work:

- Parametric schedulability analysis
- Integrate into a design-space exploration
- Study applicability to commercial microarchitectures
- “White-box” approach

Conclusions and Future Work

First general framework for
architecture-parametric timing analysis.

Future Work:

- Parametric schedulability analysis
- Integrate into a design-space exploration
- Study applicability to commercial microarchitectures
- “White-box” approach

Thank you for your attention!