# ADeLe

## Rapid Architectural Simulation of Approximate Hardware

Isaías B. Felzmann[1] Matheus M. Susin[1]

Liana Duenha[2] Rodolfo Azevedo[1] Lucas Wanner[1]

[1]Institute of Computing
University of Campinas

[2]Faculty of Computing – FACOM
Federal University of Mato Grosso do Sul

**isaias.felzmann@students.ic.unicamp.br**

Grant #2017/08015-8, São Paulo Research Foundation (FAPESP)

30th International Symposium on Computer Architecture and High Performance Computing, September 2018

# Background – Why Approximate Computing?

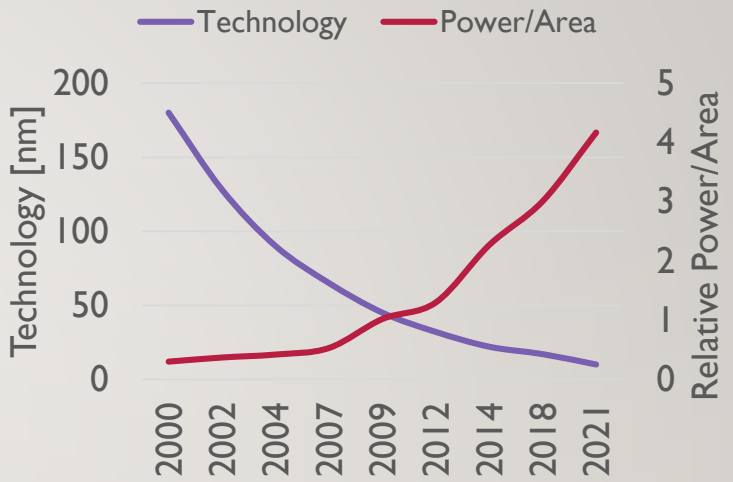
- End of Dennard Scaling:

  "The last few generations of scaling [...] seem to be difficult to scale much further." (Dennard, 2015)

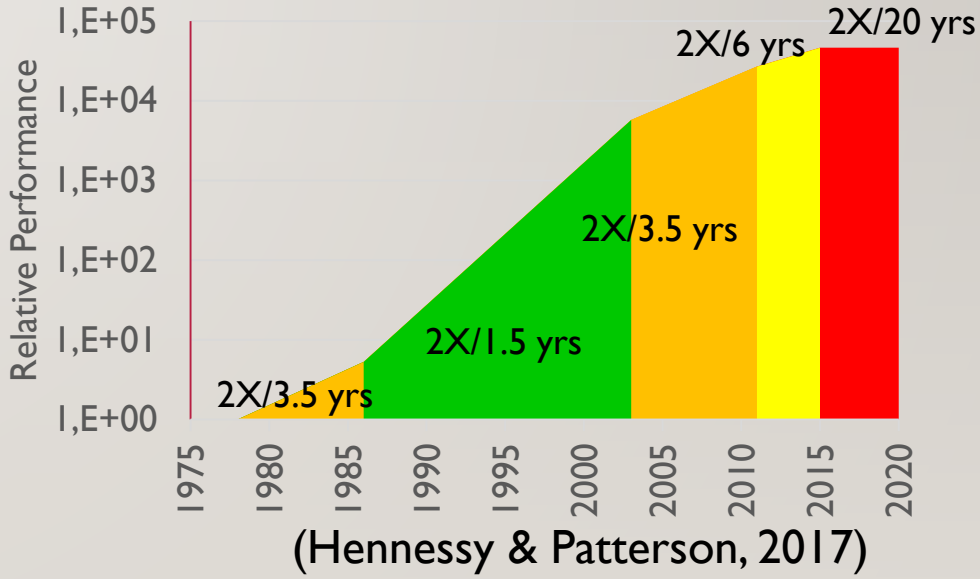
  - *Power/Energy Wall*
  - *Utilization Wall*
  - *Dark/Dimmed Silicon*

- End of Moore's Law (?)
  - No more "for free" performance improvement

- Approximate Computing trades quality for efficiency:
  - But for some (range of) applications, that is OK!
  - **How wide is this range?**

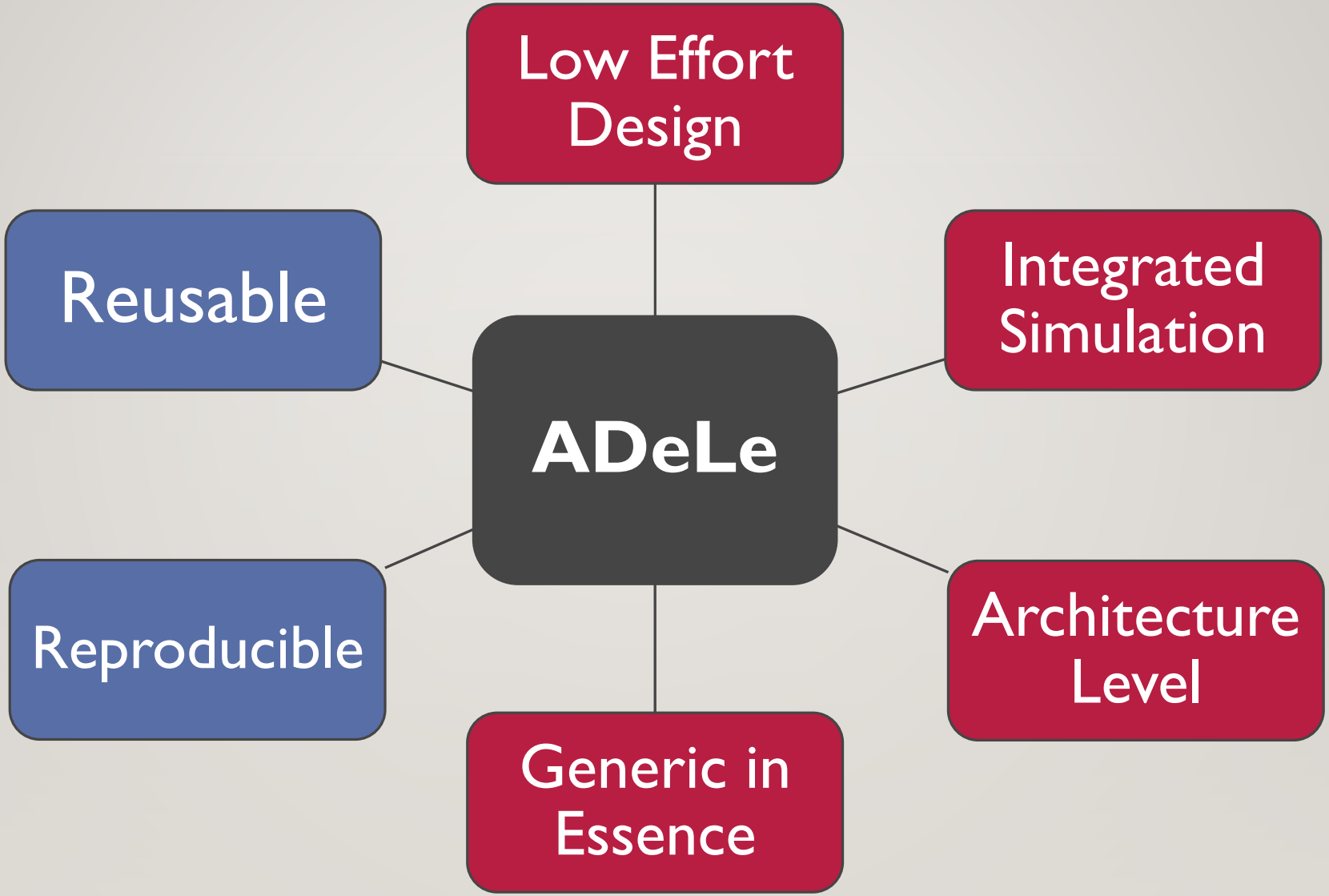



(Hennessy, 2018 based on Esmaeilzadeh *et al.*, ISCA'2011)




(Hennessy & Patterson, 2017)

# The Approximation Description Language



# Approximate Computing Design – Scenario

Dr. Alice

Bob

- Reduced precision FP operations

- HDL design (CPU and FPU)
- RTL Simulation
- Energy estimation
  - Half FPU: -30% energy

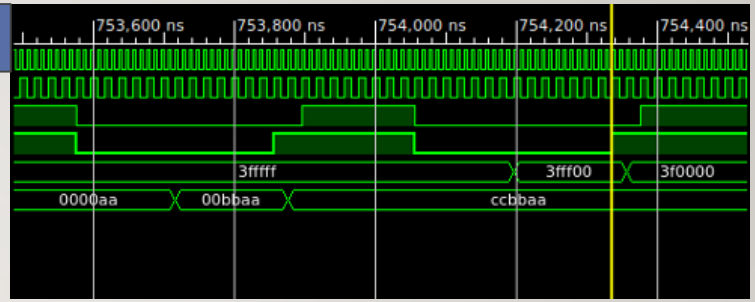
- Our paper deadline is next week!

- Need impact on applications!
- What now?

# Approximate Computing Design – Verification Alternatives

## RTL Simulation

- Very precise
- Existing frameworks
- Known tools (or are they?)
- Slow simulation
- High design effort
- High experimentation effort

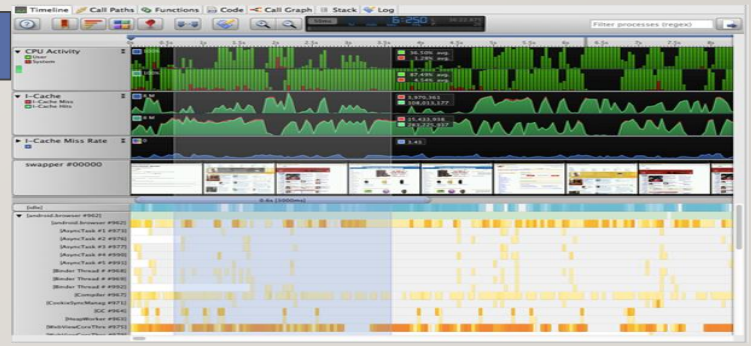



## Software Modeling / Instrumentation

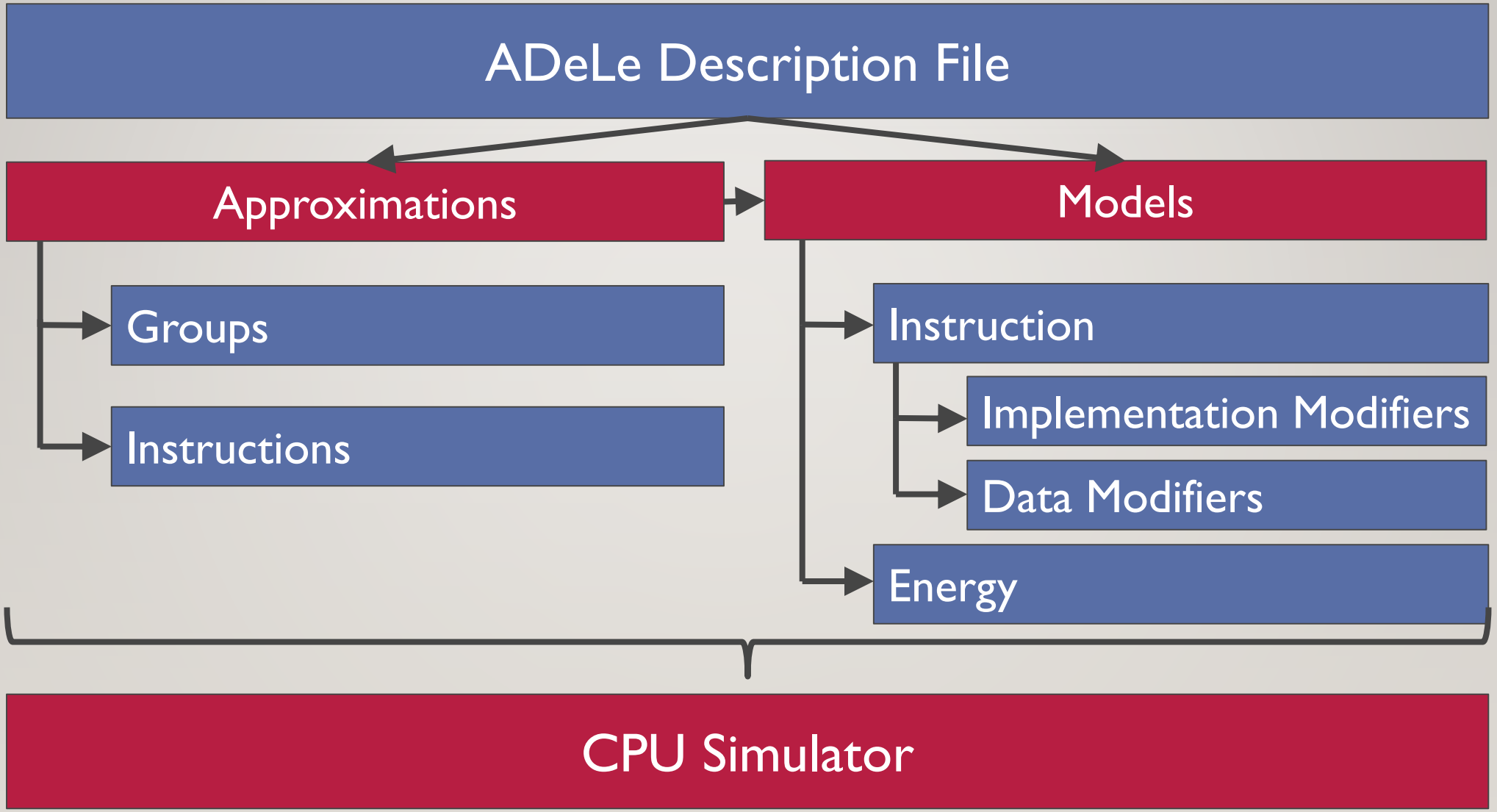
- Low effort
- Existing frameworks
- Specific to application
- No energy model
- Does not represent integration

## System Simulation

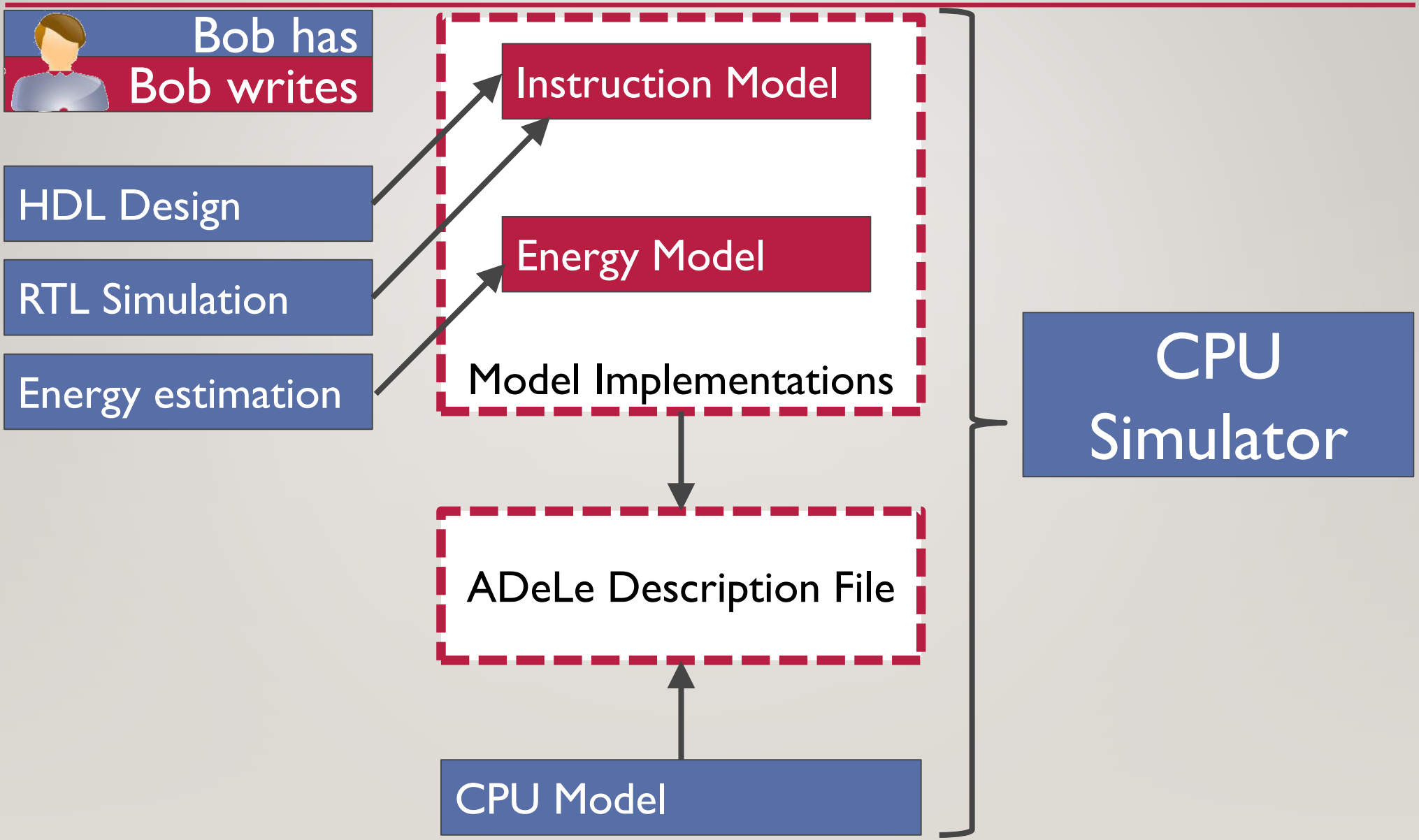
- Represents full system
- Computes energy efficiency
- ~~High effort~~
- ~~Incomplete frameworks~~



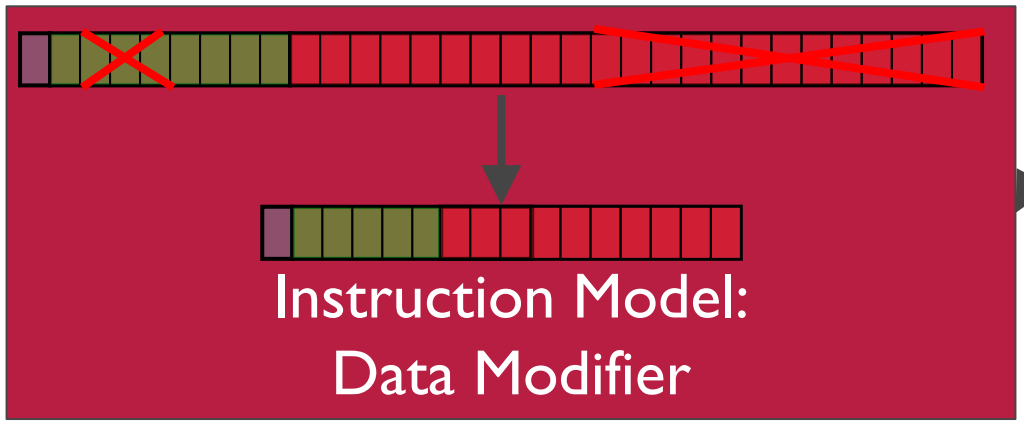
# The ADeLe Alternative



# Approximation Description in ADeLe



# Practical Example I – Half-Precision FP



Instruction Model:
Data Modifier

**FPU instructions:**
**Energy += 0.7**

Energy Model

**Other instructions:**
**Energy += 1.0**

Energy Model

Model Implementations

```
DM Float2HalfPrecision();
EM HalfFPU();
EM Default();

energy = Default();

group FP {/*instructions*/}

approximation HALF_FP {
  group FP {
    regbank_read =
      Float2HalfPrecision();
    regbank_write =
      Float2HalfPrecision();
    energy =
      HalfFPU();
  }
}
```

ADeLe Description File

# Practical Example II – Approximate Multiplication

**EvoApprox8B**
**(Mrazek et al., DATE'2017)**
**Library of mutipliers**
Instruction Model:
Implementation Modifier

**Multiplication instructions:**
**Energy += x (<1.0)**
Energy Model

**Other instructions:**
**Energy += 1.0**
Energy Model

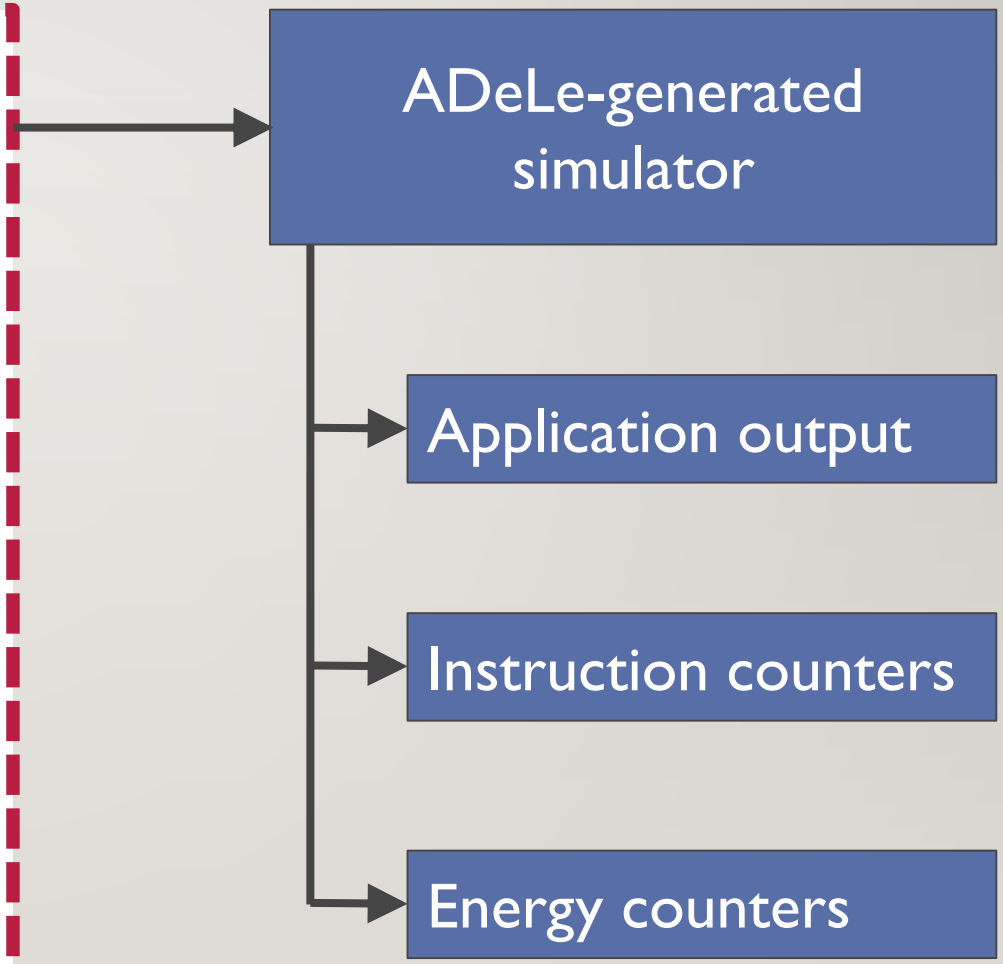
Model Implementations

```
IM EvoApprox8B_Mult303()
EM ApproxMult303();
EM Default();

energy = Default();


approximation EVOAPPROX8B {
  instruction mult {
    alt_behavior =
      EvoApprox8B_Mult303();
    energy =
      ApproxMult303();
  }
}
```

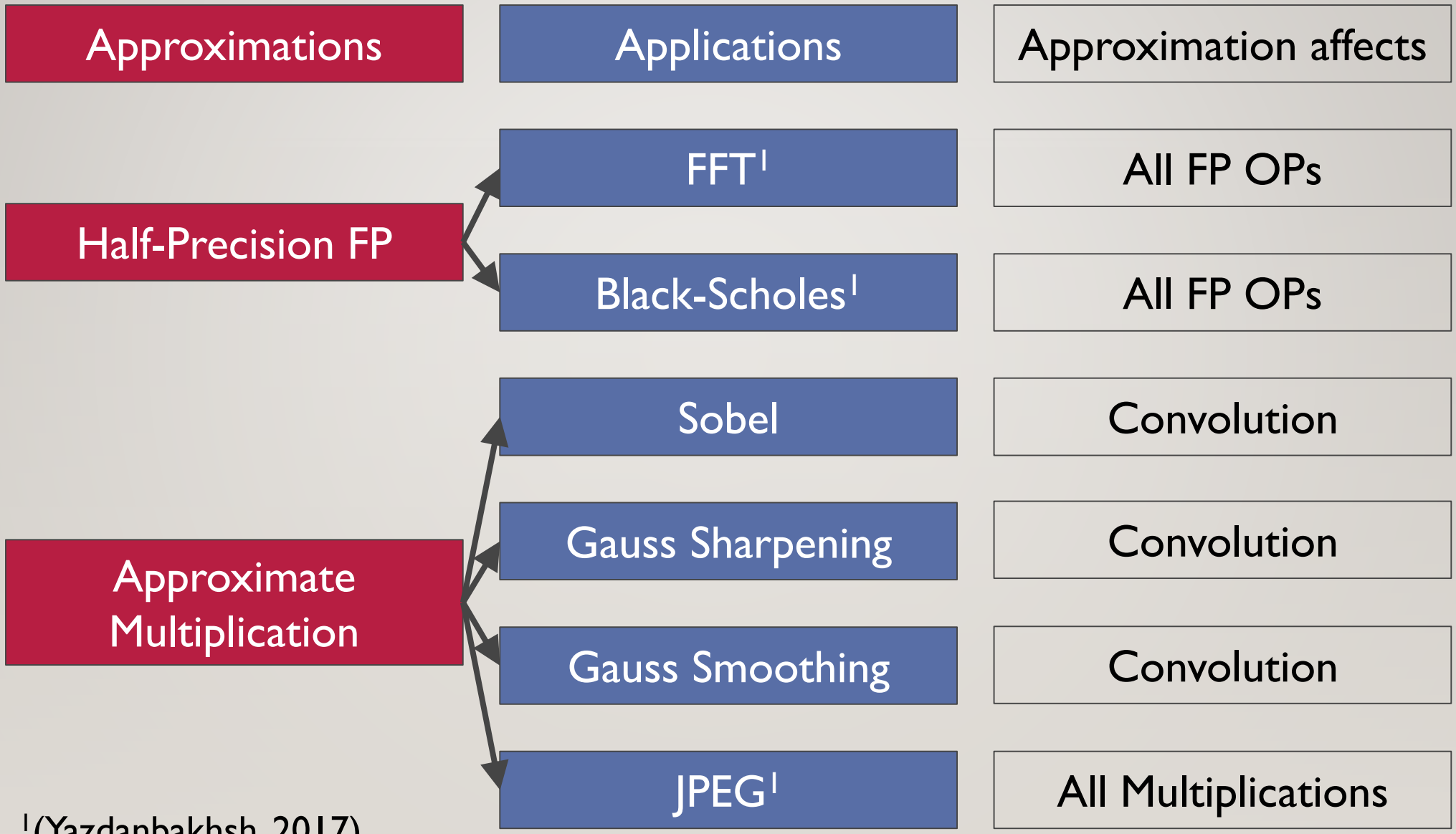
ADeLe Description File

# RUNNING AN APPLICATION

```
#include "adele_iface.h"

adele_activate(HALF_FP);

for (i = 0; i < n; i++) {
  sum = 0.0;
  for (j = 0; j < n; j++) {
    for (k = 0; k < n; k++) {
      sum += A[i][k] + B[k][j];
    }
    C[i][j] = sum;
  }
}

adele_deactivate(HALF_FP);
```

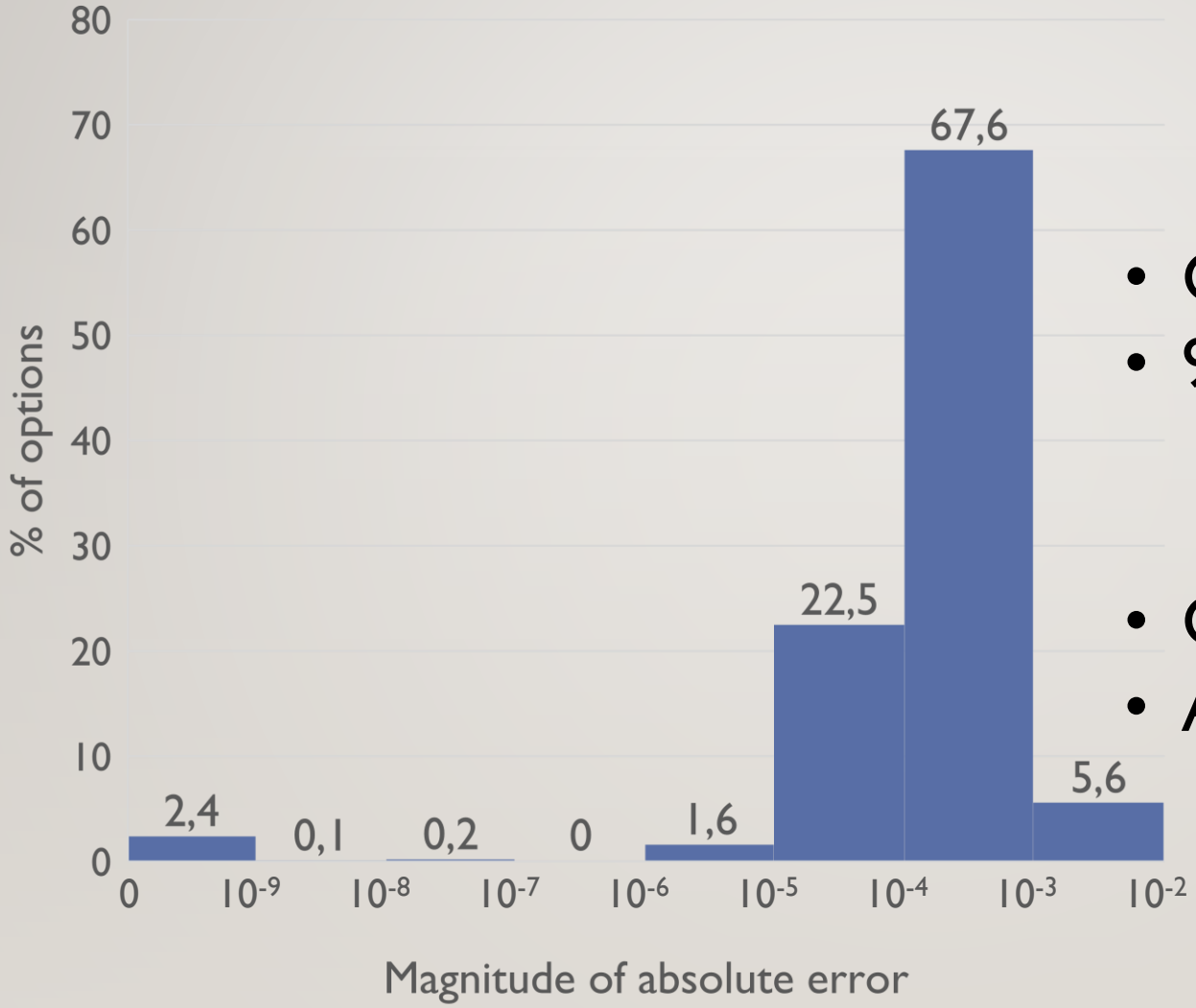


# VERIFYING ADELE




[1](Yazdanbakhsh, 2017)

# Verification Results – Half-Precision Floating-Point



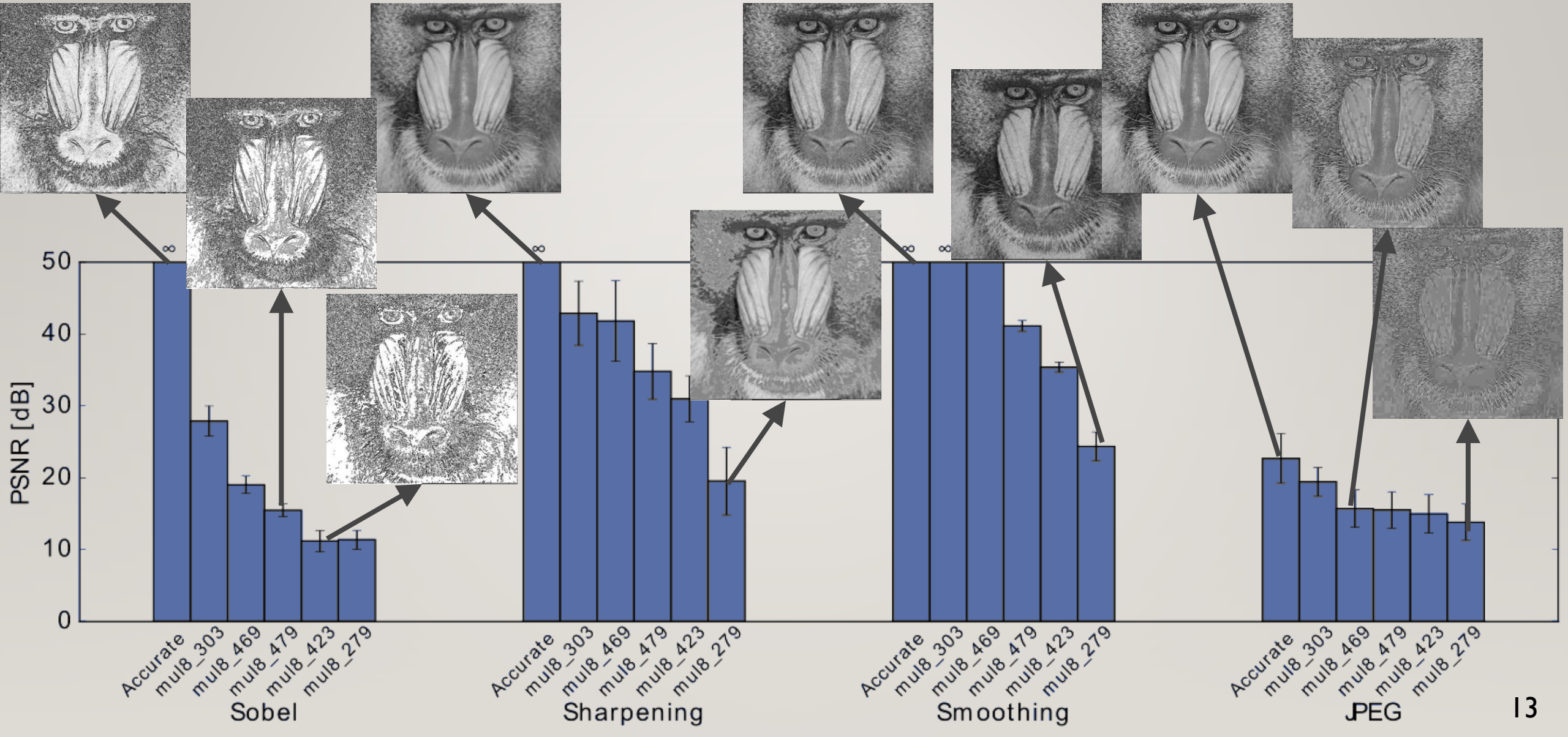

**Black-Scholes**

- Covers 44.6% of instructions
- 94.4% below $10^{-3}$

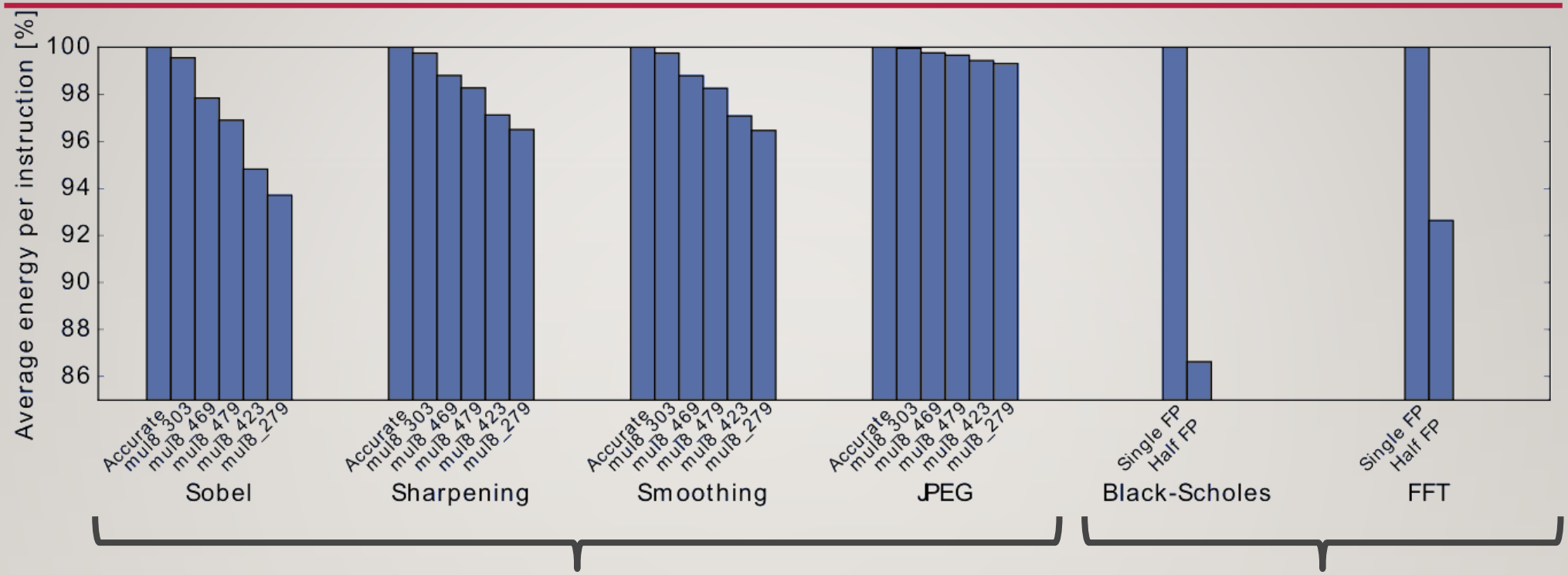
**FFT**

- Covers 24.8% of instructions
- All within ±1% error

# VERIFICATION RESULTS – IMAGE PROCESSING



# VERIFICATION RESULTS – ENERGY




- Overhead from loop control
- More multiplication-intensive kernels can benefit further

- Higher coverage, higher savings (Amdahl's law)

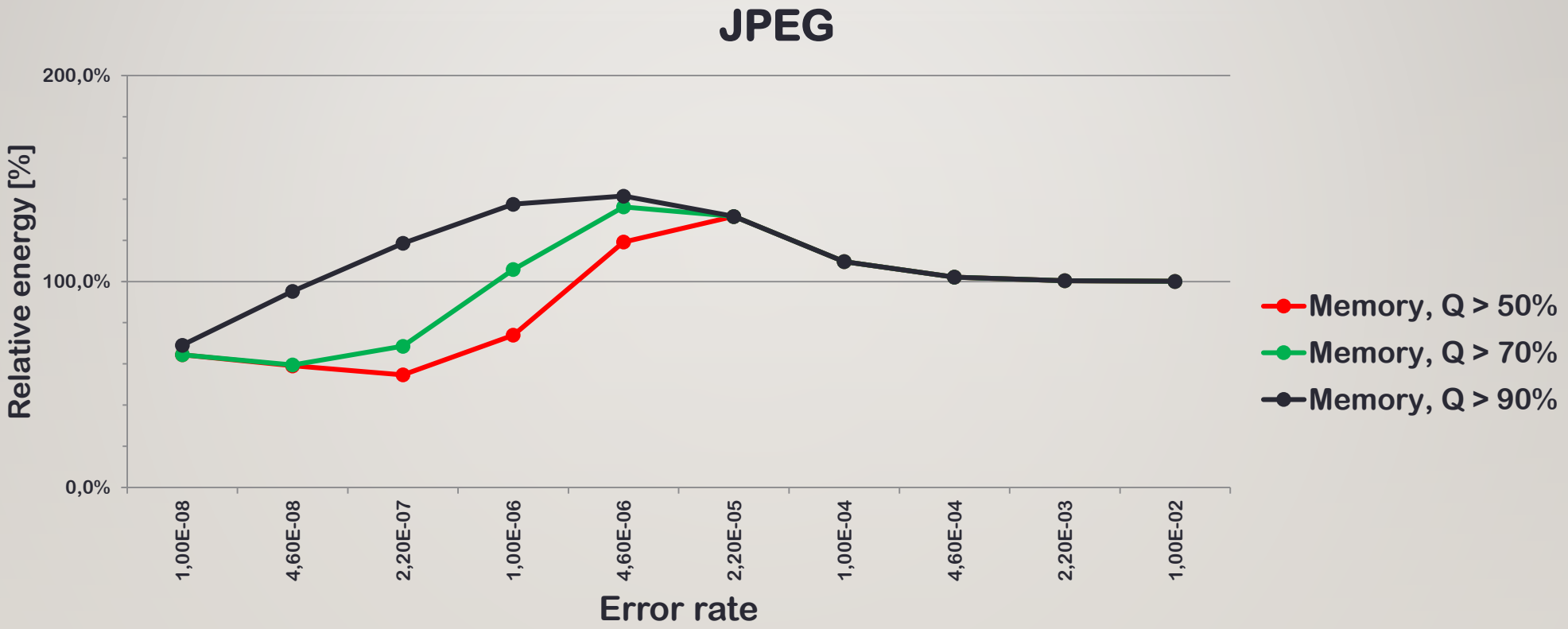
Potentially higher benefits if part of the critical path
Modeled in ADeLe using different Energy Model

# Final Remarks

- How wide is the range of applications that benefit from Approximate Computing?
  - ADeLe allows for design space exploration with minimal effort
- ArchC integration (the VArchC framework) allows for the representation of virtually any CPU model.
- Results are easily reproducible if models are disclosed.
  - And extended to other applications/architectures.

# Scenes of our Next Episode

How do applications behave under influence of approximate memories?




(Felzmann *et.al.*, WSCAD'2018)

# Thank You!

**varchc.github.io/sbac**

**isaias.felzmann@students.ic.unicamp.br**