DATE 16
DESIGN, AUTOMATION & TEST IN EUROPE
14 – 18 March, 2016 · ICC · Dresden · Germany
The European Event for Electronic System Design & Test

Fidelity of native-based performance models for Design Space Exploration

Fernando Herrera and Eugenio Villar

University of Cantabria

UC UNIVERSIDAD DE CANTABRIA

Contrex

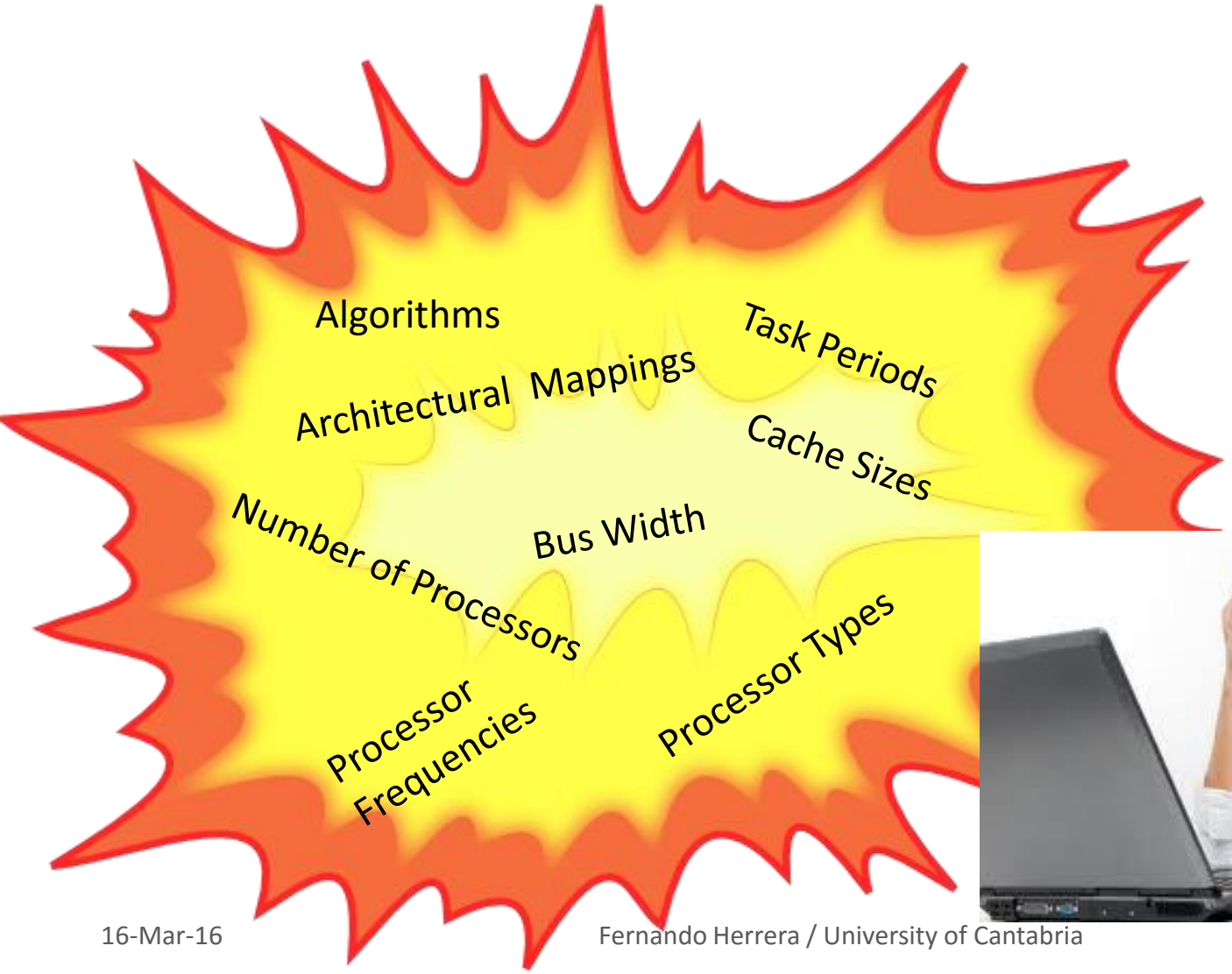
COOPERATION

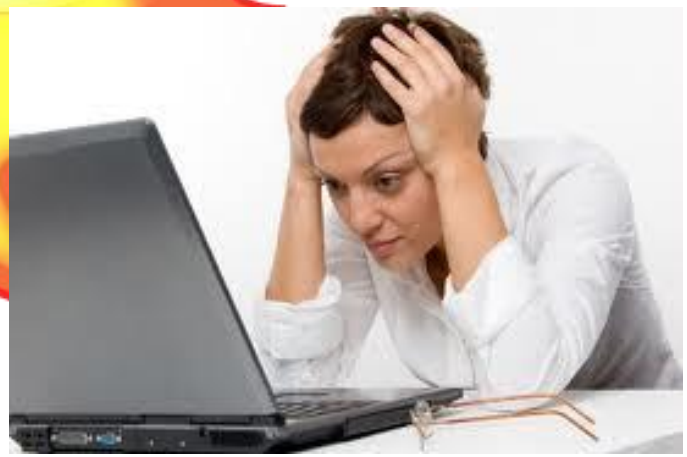
Outline

- DSE
- Fidelity in DSE
- Estimation Technologies: Native Simulation
- Fidelity in Native Simulation
- VIPPE
- Conclusion

Design Space Exploration

- **Crucial Activity in Electronic System-Level Design!!**





Exploration Space: Size Example

- **# solutions:**
  - **5 applications, 2 algorithms on app1, 2 algorithms on app2, 4 period combinations, 1-4 processors, 3 processor types, 3 frequencies, all possible application mappings**
  - **Design Space: 2x2x4x3x(3^4)x(3^5)=944784**
- **~1 million solutions**
- **5 days analysis:**

  → 200.000 solutions/day → >23 alternatives/sec

DSE requirements

- **Exploration speed**
- **“Sufficient” Accuracy: Fidelity in DSE context**

Fidelity

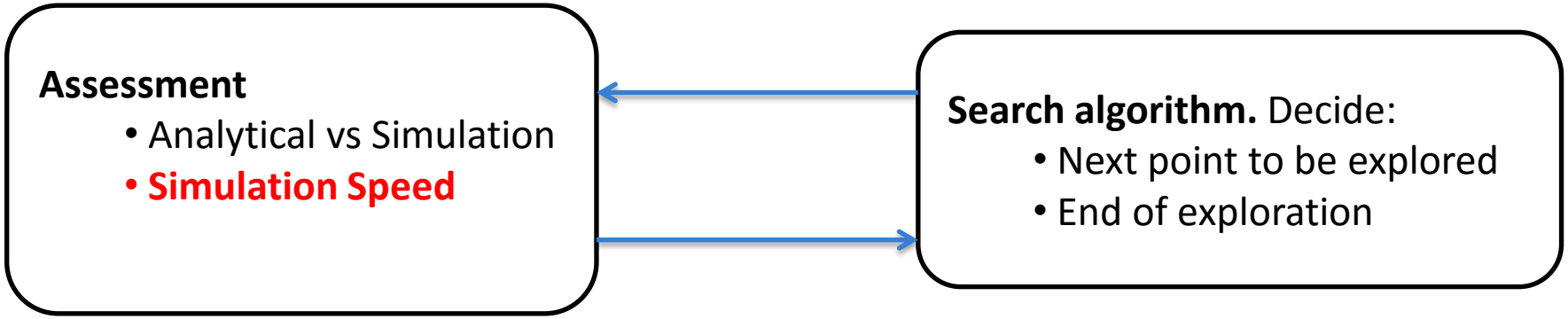
Report from the Fidelity Implementation Study Group

D.C Gross (Boeing)

"The degree to which a model or simulation reproduces the state and behavior of a real world object or the perception of a real world object, feature, condition, or chosen standard in a measurable or perceivable manner; ***a measure of the realism of a model or simulation****; faithfulness.*

Fidelity should generally be described ***with respect to the measures, standards or perceptions*** *used in assessing or stating it. See* ***accuracy****, sensitivity, precision, resolution, repeatability, model/simulation validation".*

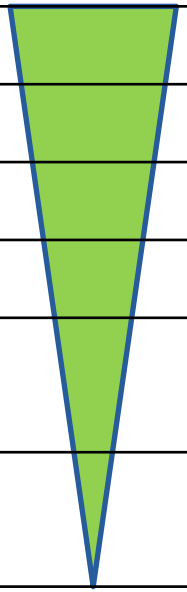
DSE requirements (I): Exploration Speed



- **Assessment transition**
  - Data Exchange with exploration algorithm
  - Configure the model to be explored

Simulation-based Performance Assessment Technologies

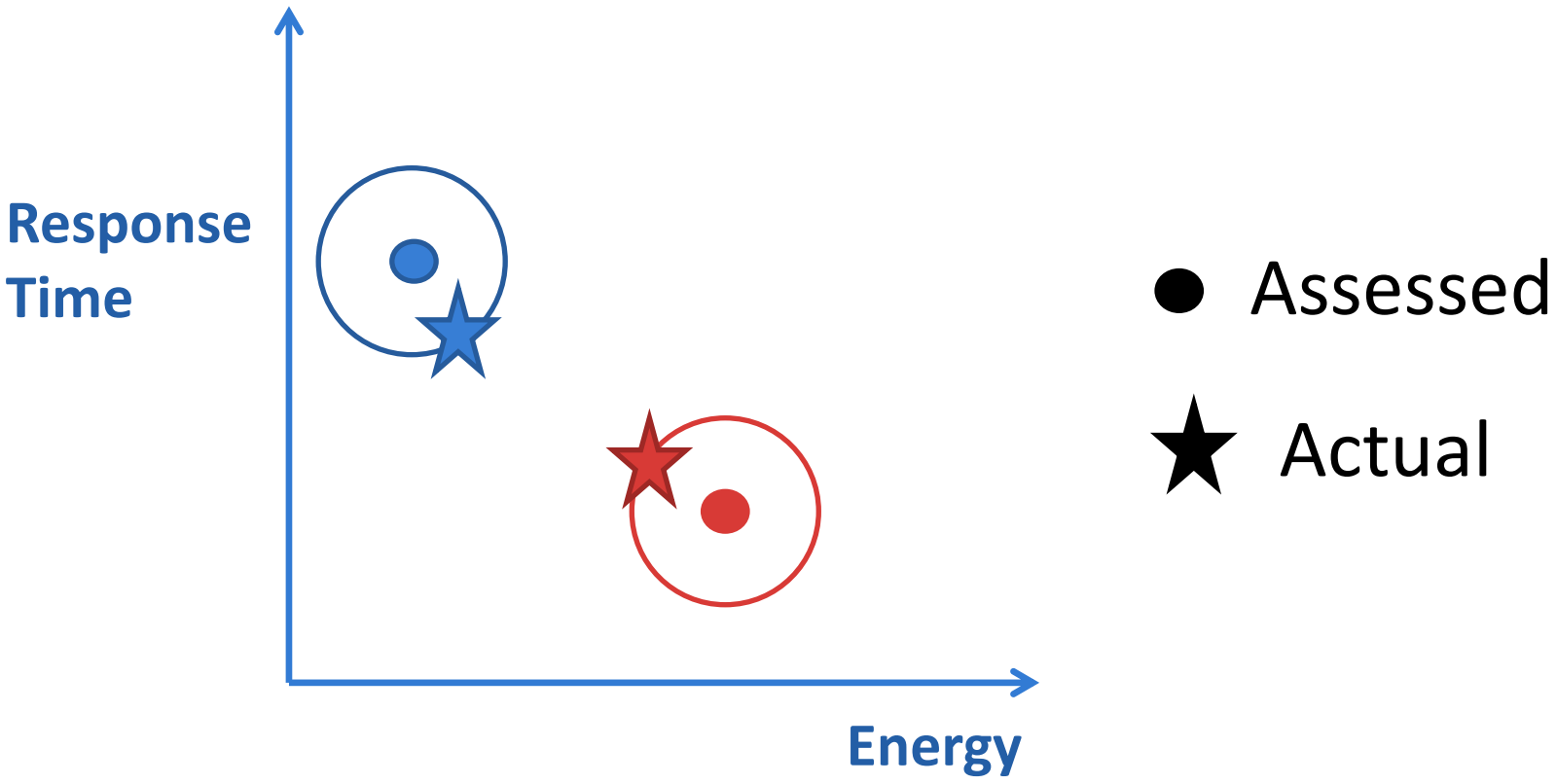
	Speed Up	Accuracy Degradation
Functional Execution	100000	?
Timed native co-simulation	10000	
Timed binary translation (QUEMU+, OVP+)	1000	
ISS (instructions)	100	
ISS with microarch. (cycle accurate)	10	
Pin&Cycle Accurate (HDL simulator)	1	



- **Simulation-based Performance Assessment Technologies**

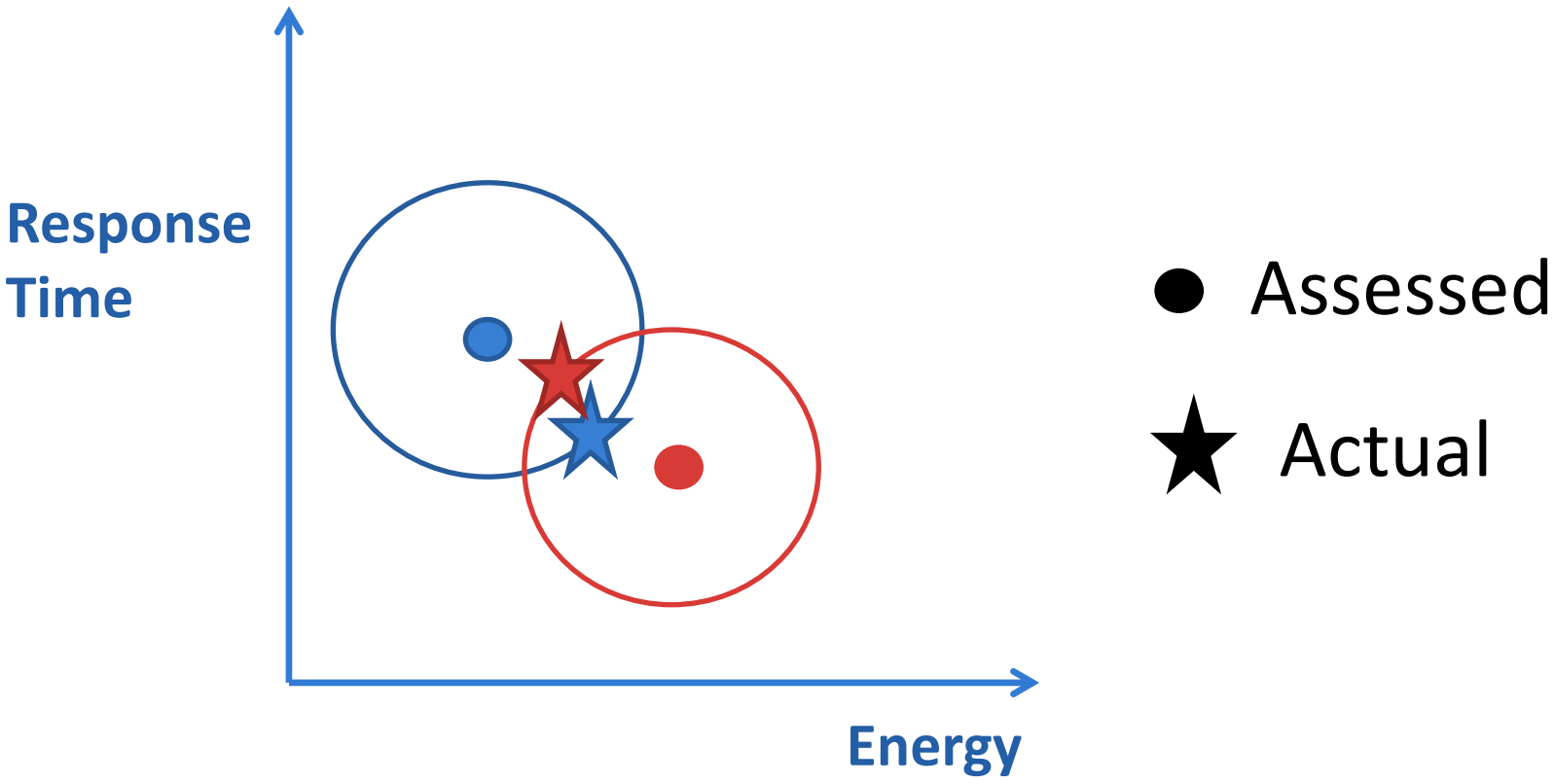
DSE requirements (II)

- **Sufficient Accuracy**



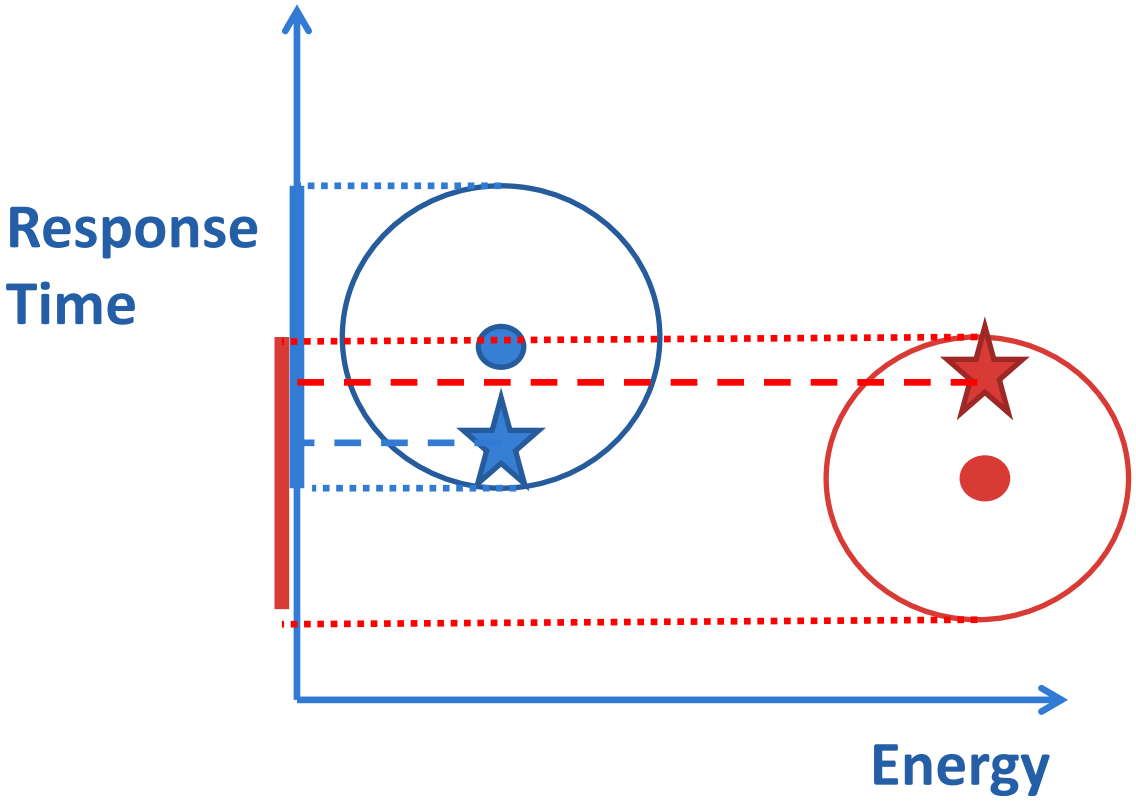
DSE requirements (II)

- **Insufficient Accuracy**



DSE requirements (II)

- **Insufficient Accuracy**



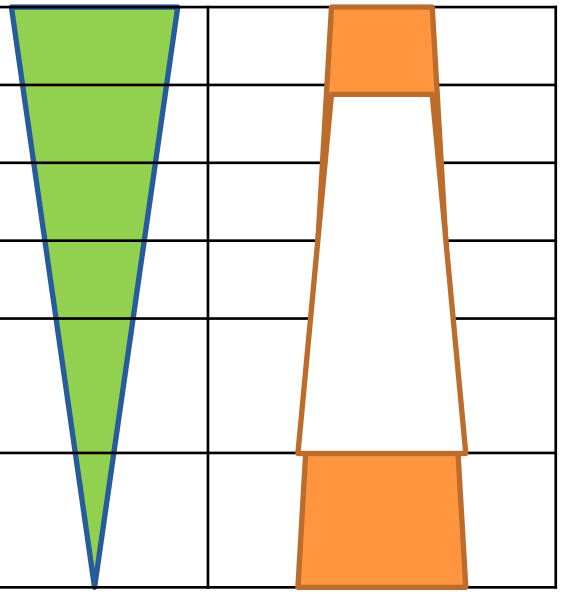
● Assessed

★ Actual

Fidelity in DSE (as accuracy)

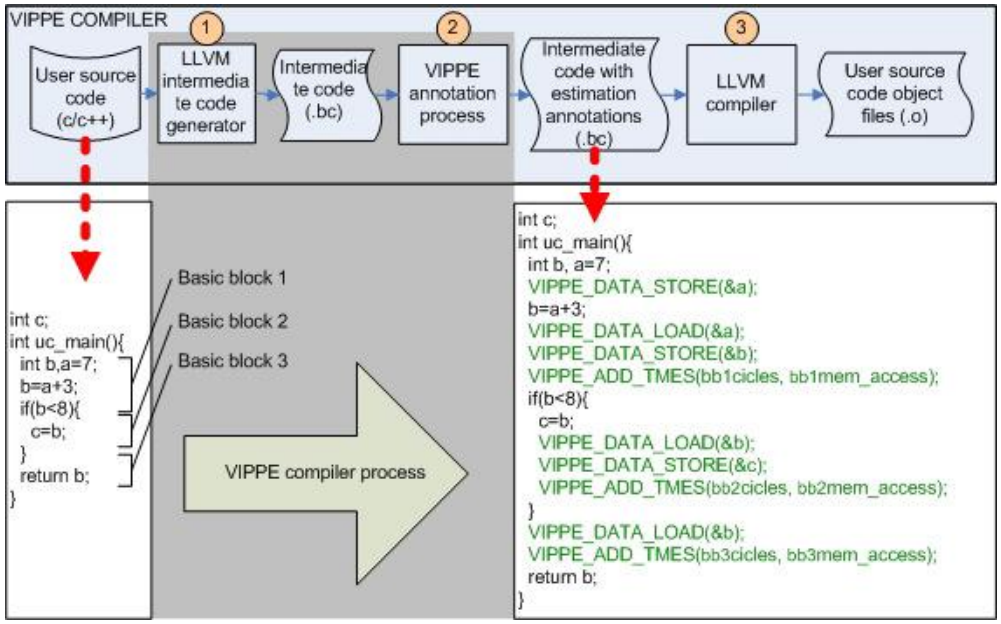
- **Depends on the amount of solutions and the constellation amplitude (= density)**
- **One (Our) Criteria: <10% vs instructions accurate technologies**

	Speed Up	Accuracy Degradation
Functional Execution	100000	
Timed native co-simulation	10000	
Timed binary translation (QUEMU+, OVP+)	1000	
ISS (instructions)	100	
ISS with microarch. (cycle accurate)	10	
Pin&Cycle Accurate (HDL simulator)	1	



Native Simulation

- Instrument the source code
  - For performance cost accounting
  - Generating memory access traces
- Simulation speed: ~ (native program speed) * α (α <2)
  - Overhead of accounting and trace generation



Fidelity in Native Simulation

- In using the **same source code**

- In the **assessment of performance**, in the placing of events in the simulated time, considering the effects and penalties of:
  - SDK effect (compiler)
  - RTOS (if any)
  - hardware architecture

- **Not (or not necessarily) in:**
  - Placement of events on the emulated time axis (time decoupling)
  - the behaviour of sw platform and details relevant to SW development, e.g., in detecting every type of ilegal memory access in the target platform.

Native Simulation: Fidelity Challenges

- **Input code:**
  - custom parsing & reconstruction vs LLVM
  - Object/Legacy Libraries [Henkel, CODES+ISSS'12]
- **SDK (Compiler) Complexity**
  - Optimizations [Rosenstiel, DAC11] [Henkel, DATE'12]
  - Target dependent headers
- **Platform Complexity:**
  - Target Processor
    - Word width, ISA [Posadas, 2006]
    - Internal architecture [Gerslauer, CODES'13][Rosenstiel, ESWEEK'14]
  - RTOS [Posadas, 2006][Gerlauer, DATE'11,ACM TECS 2014]
  - Cache model [Posadas,ASP-DAC'11][Rosenstiel, DATE 12] [Gerslauer, ESLyn'13]
  - Memory hierarchies [Posadas,SIES'10]
  - Platform architecture [Posadas, 2009][Rosenstiel, DATE'11]
  - Cache coherence

Parallel Native Simulation

- **P: # host cores, M: # target cores, T: # application(s) threads**
- **Exploit the underling parallelism of the host machine simulating each processing element in one core**
  - **Potential speed up: min(P,M,T)**
  - >1 as long as M>1 ^ P>1 ^ T>1
  - Also in other technologies
- **Thread-based parallelization (VIPPE)**
  - **Potential speed up: min(P,T)**
    - **>1 as long as P>1 ^ T>1, even if M=1**
    - **Breaks the M speed-up wall, as gain can be P if P > M ^ T>M**
  - **Challenges cache estimation**

VIPPE

- Parallel native simulation
- Compilation branches: LLVM and source compiler
  - Optimizations
- Platform modelling
  - Processor modelling: armv7, sparc, LEON, microblaze
  - I&D Cache model
  - Heterogeneous Multi-core Platform Architecture
- Connection to SystemC
- Performance reports: time, energy, power
- GUIs

Conclusions

- **Native Simulation: key simulation technology for simulation-based Design Space Exploration of complex embedded systems**
- **Fidelity required**
- **Fidelity requires information regarding the target processor, the cross-development kit, and the target architecture.**
- **VIPPE: last generation of native simulation tool with Fidelity for Design Space exploration**
- **More soon in http://vippe.teisa.unican.es**