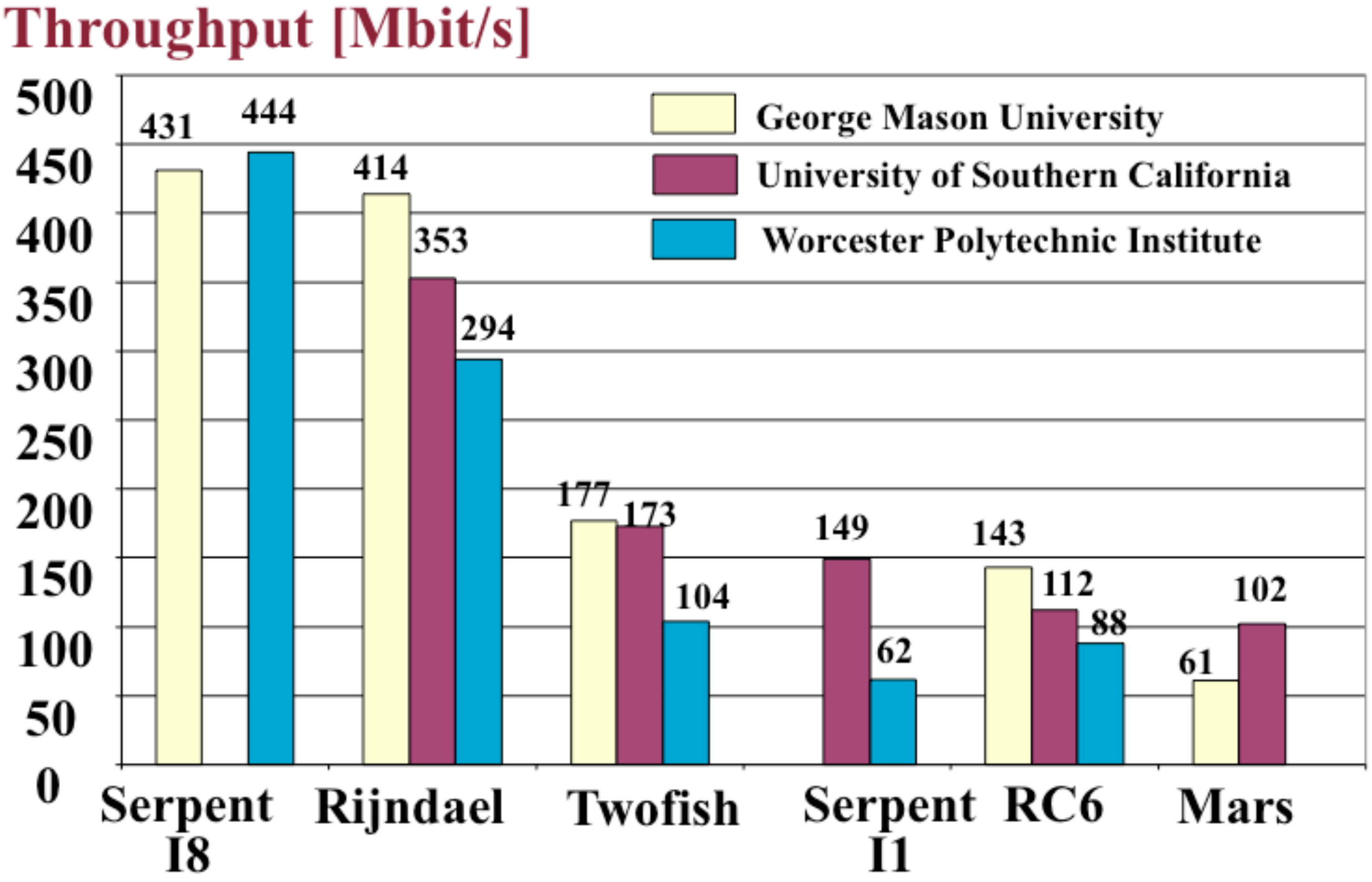
Benchmarking of Cryptographic Hardware

Kris Gaj
George Mason University

Lessons from the Past

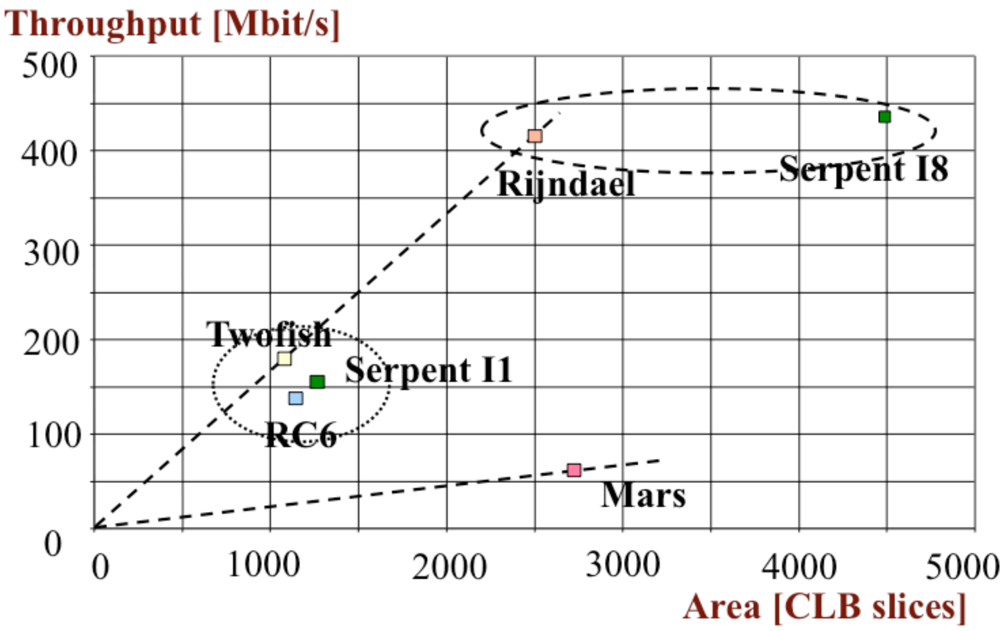
Various groups tend to choose independently the same (or similar) devices and tools

Round 2 of AES contest, 1999-2000: Xilinx Virtex 1000, Xilinx ISE



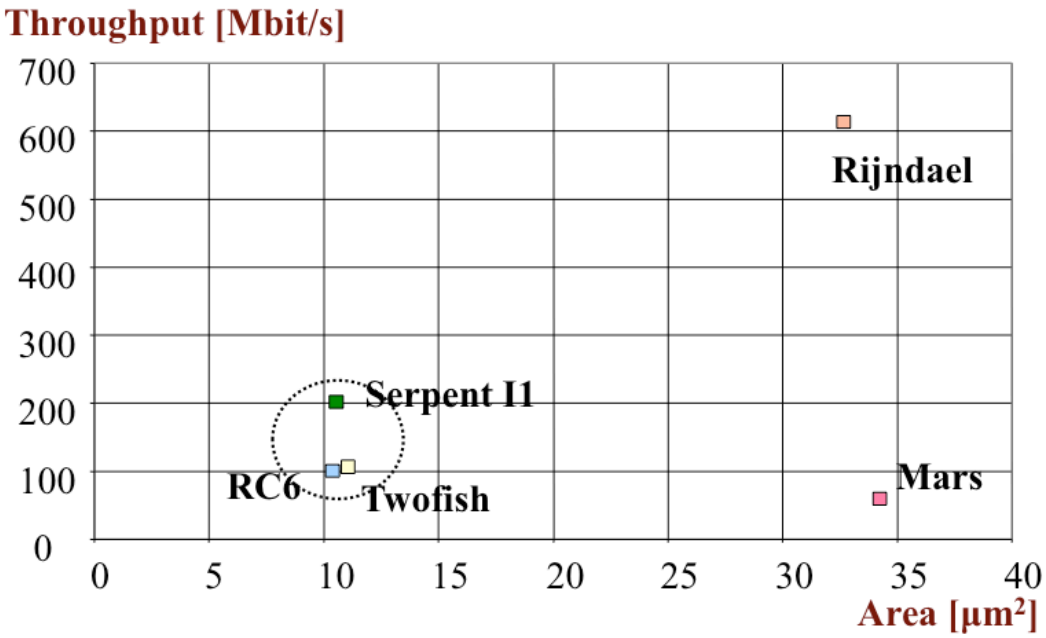
Results for ASICs match very well results for FPGAs, and are both very different than software

FPGA



GMU+USC, Xilinx Virtex 1000, **CLB Slices only**

ASIC

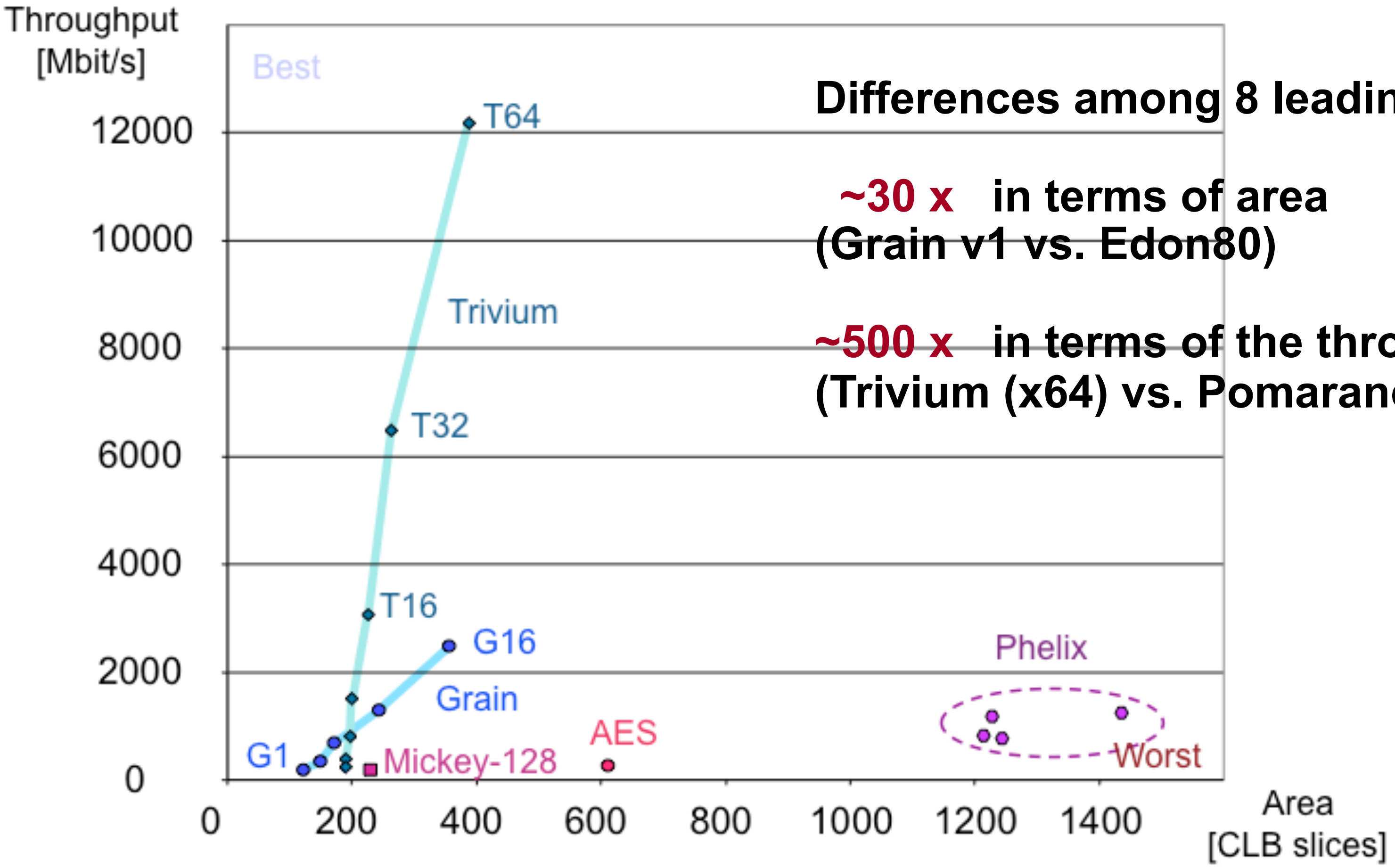


NSA Team, ASIC, 0.5 µm CMOS

Serpent fastest in hardware, slowest in software

Differences in hardware efficiency of cryptographic algorithms (even of the same type) are very significant

eSTREAM contest, 2007-2008: FPGA, Xilinx Spartan 3



Differences among 8 leading candidates:

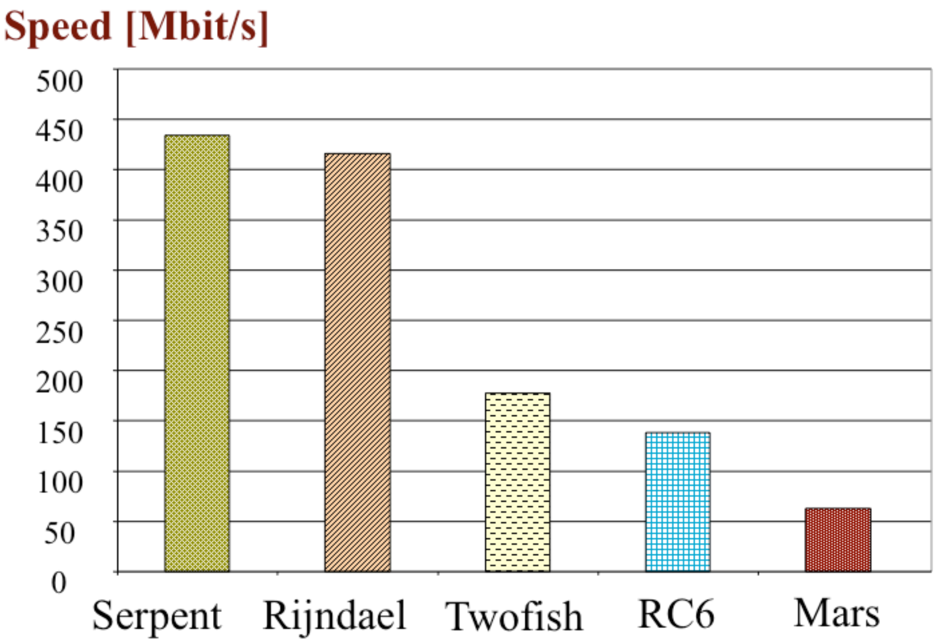
~30 x   in terms of area
(Grain v1 vs. Edon80)

~500 x   in terms of the throughput to area ratio
(Trivium (x64) vs. Pomaranch)

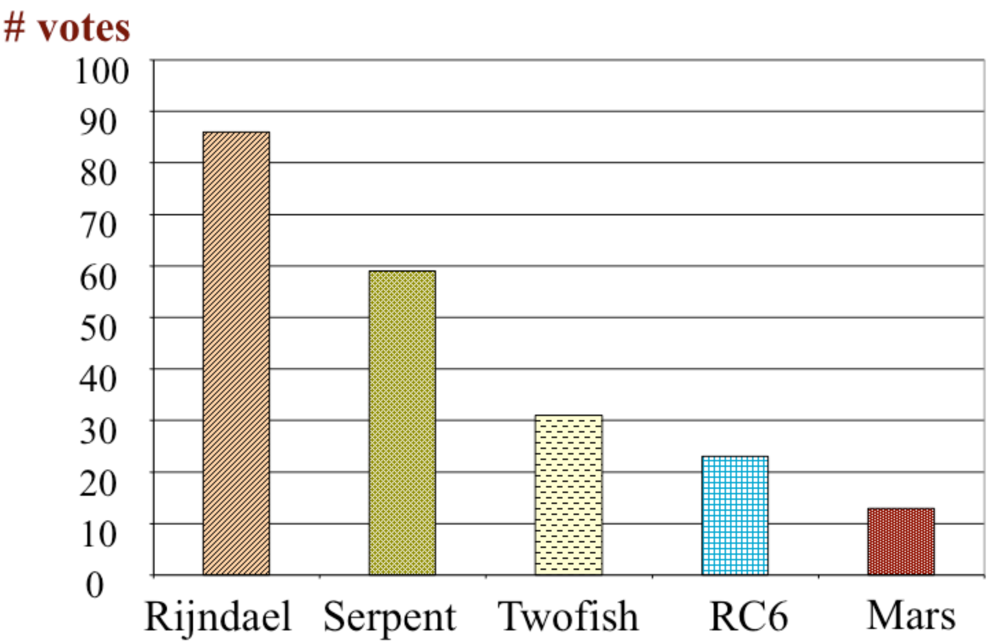
Hardware results matter!

Round 2 of AES Contest, 2000

Speed in FPGAs



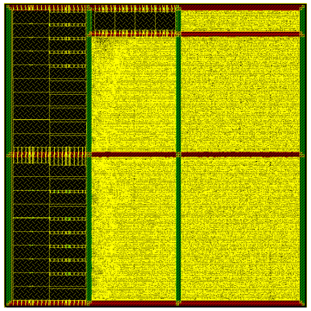
Votes at the AES 3 conference

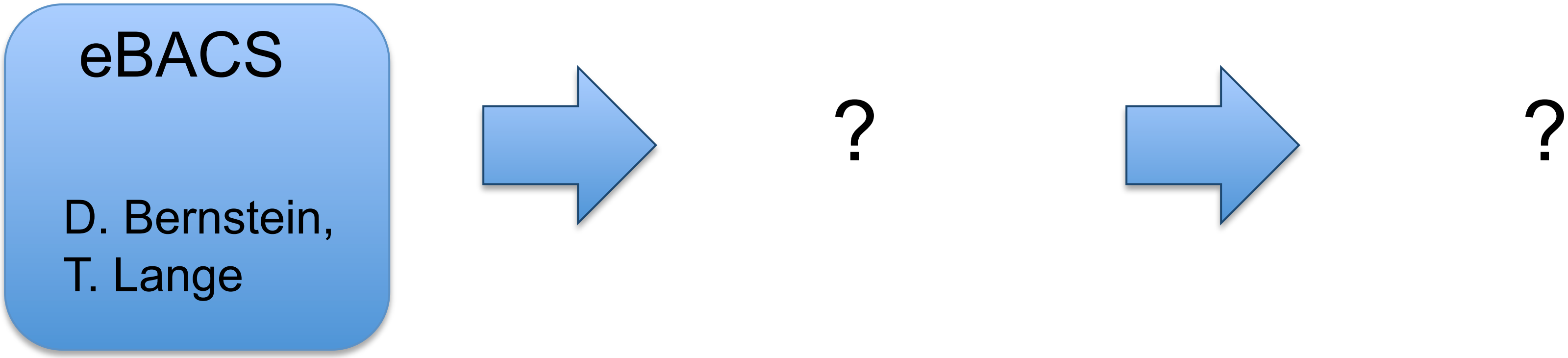


Plans for the Future

Modern Benchmarking

Software **FPGAs** **ASICs**





eBACS

D. Bernstein,
T. Lange

?

?

Our Solution

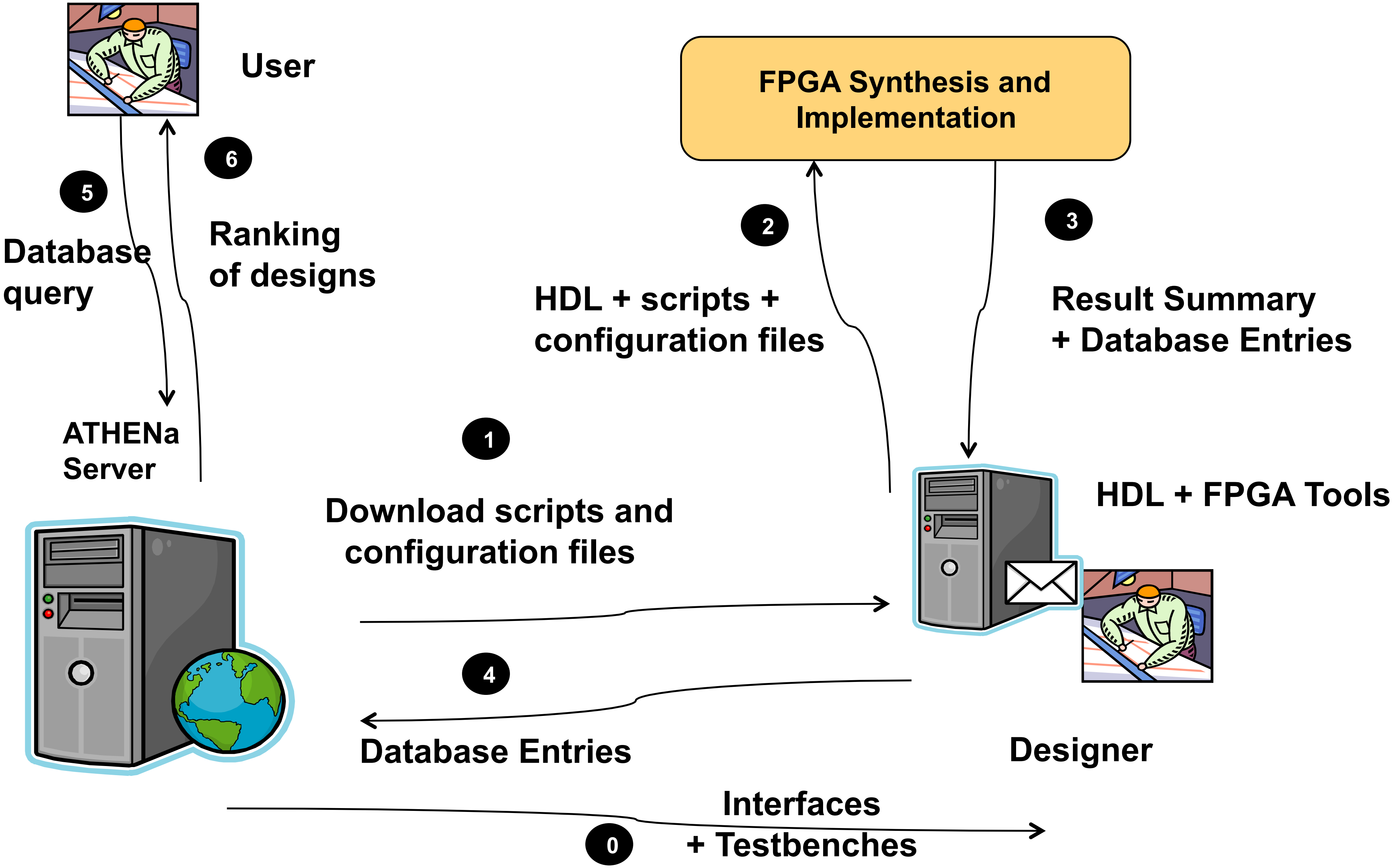
ATHENa – **A**utomated **T**ool for **H**ardware **E**valuatio**N**

Set of scripts written in Perl aimed at an
AUTOMATED generation of
OPTIMIZED results for
MULTIPLE hardware platforms

Currently under development at
George Mason University.

More details about the project at
http://cryptography.gmu.edu/athena

Basic Dataflow of ATHENa



User

FPGA Synthesis and Implementation

5 Database query

6 Ranking of designs

2 HDL + scripts + configuration files

3 Result Summary + Database Entries

ATHENa Server

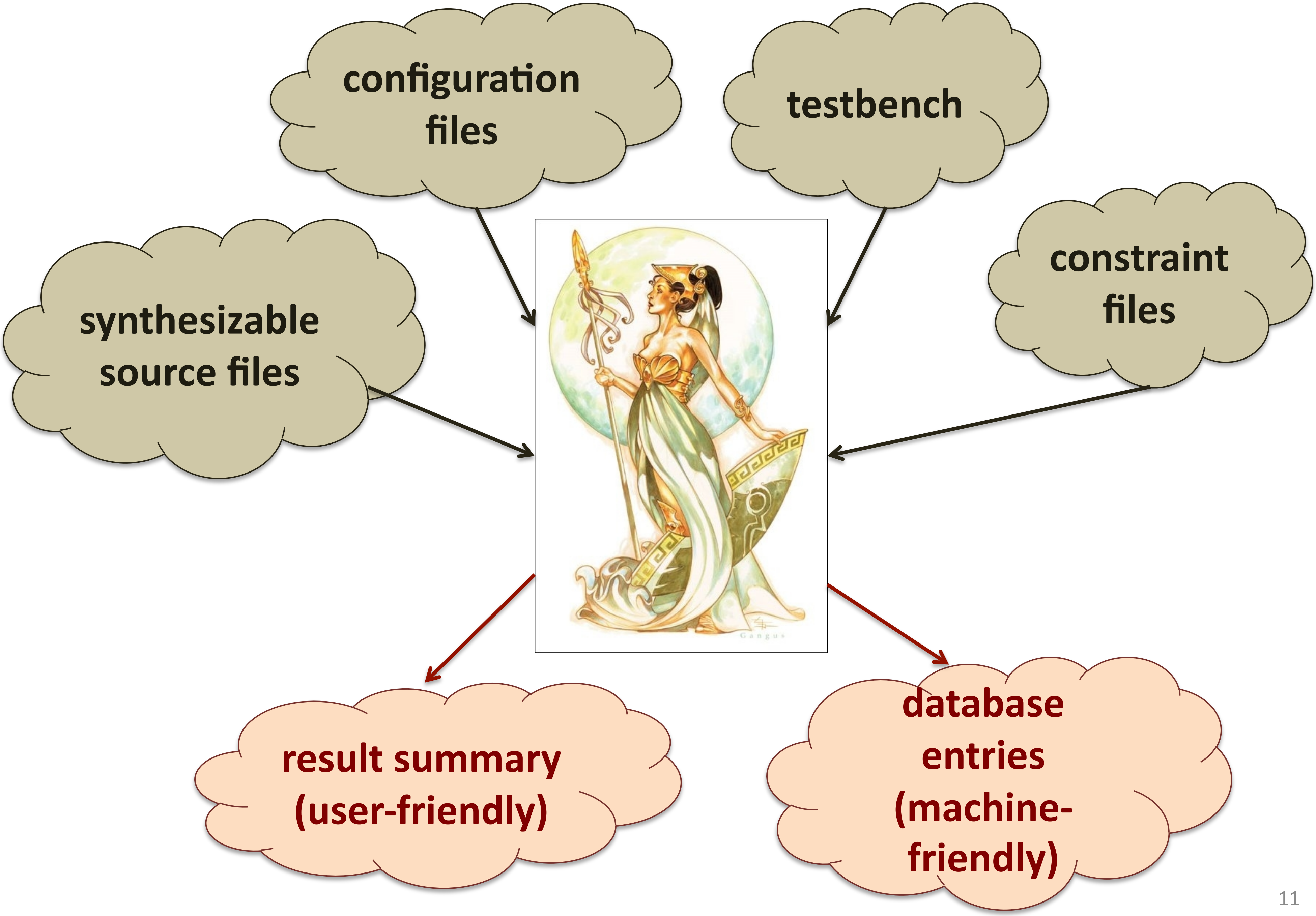
1 Download scripts and configuration files

HDL + FPGA Tools

4 Database Entries

Designer

0 Interfaces + Testbenches



configuration files

testbench

synthesizable source files

constraint files

result summary (user-friendly)

database entries (machine-friendly)

ATHENa Major Features (1)

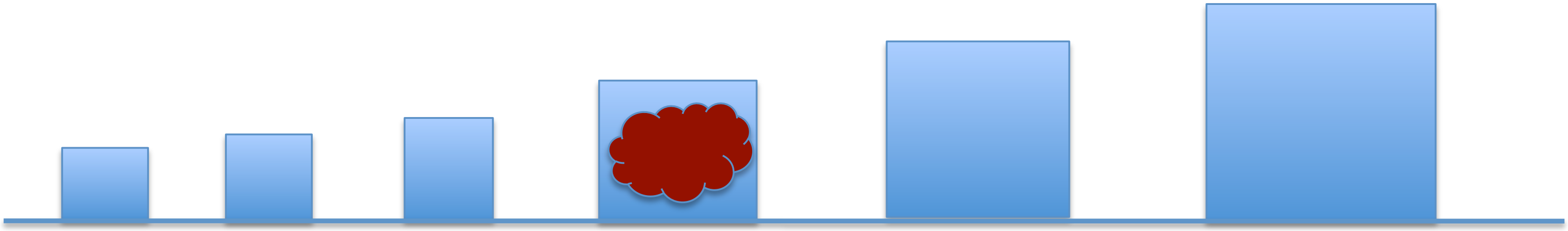
- synthesis, implementation, and timing analysis in the **batch mode**
- support for devices and tools of **multiple FPGA vendors**:

XILINX. ALTERA. Actel POWER MATTERS

- generation of results for **multiple families** of FPGAs of a given vendor

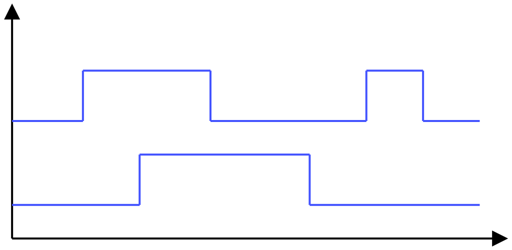
VIRTEX VIRTEX-II PRO SPARTAN FPGA SPARTAN-II FPGA

- automated choice of a **best-matching device** within a given family



ATHENa Major Features (2)

- **automated verification** of the design through simulation in the batch mode



OR

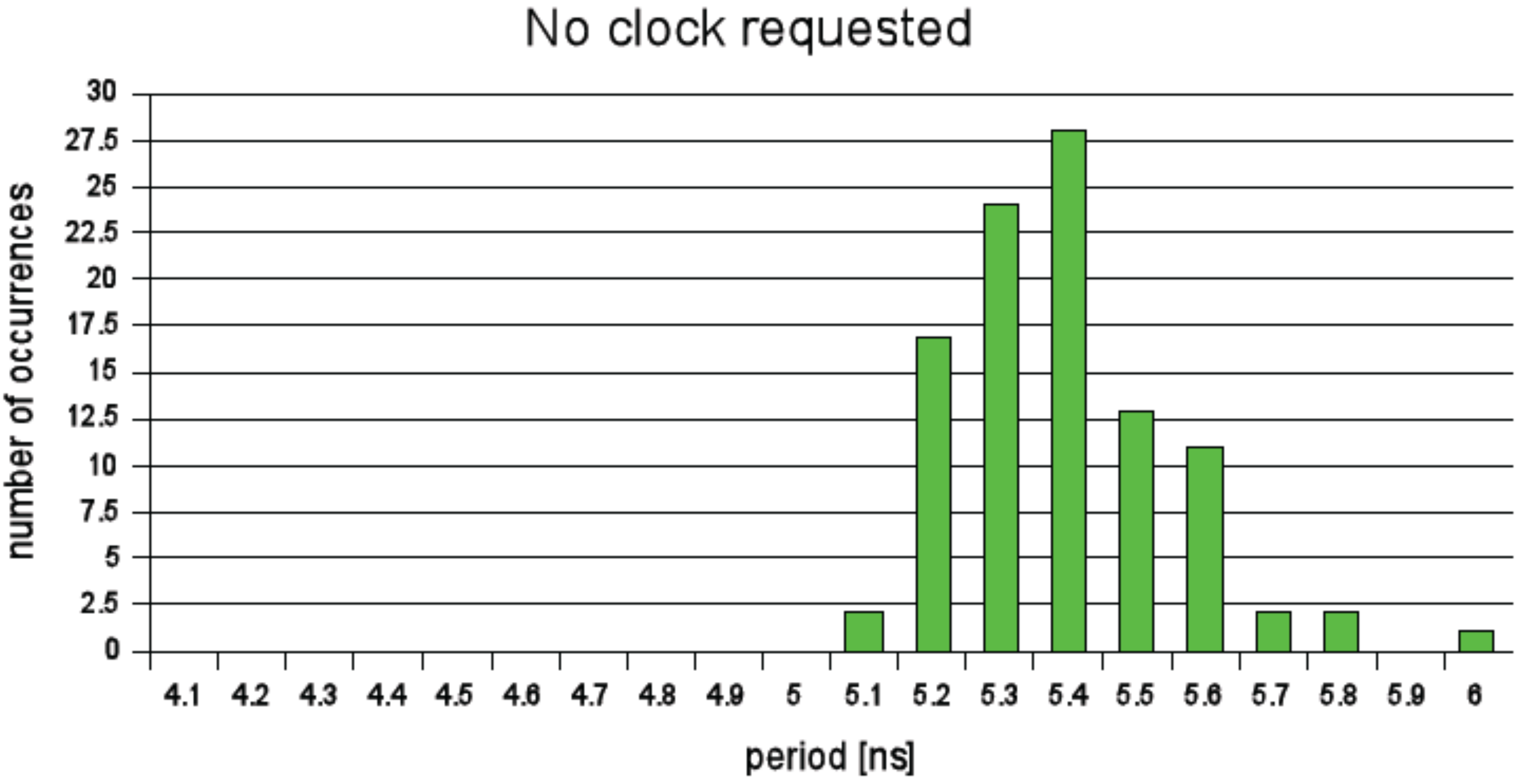
- **exhaustive search** for optimum options of the tools

- **heuristic optimization algorithms** aimed at maximizing selected performance measures (e.g., speed, area, speed/area ratio, power, cost, etc.)

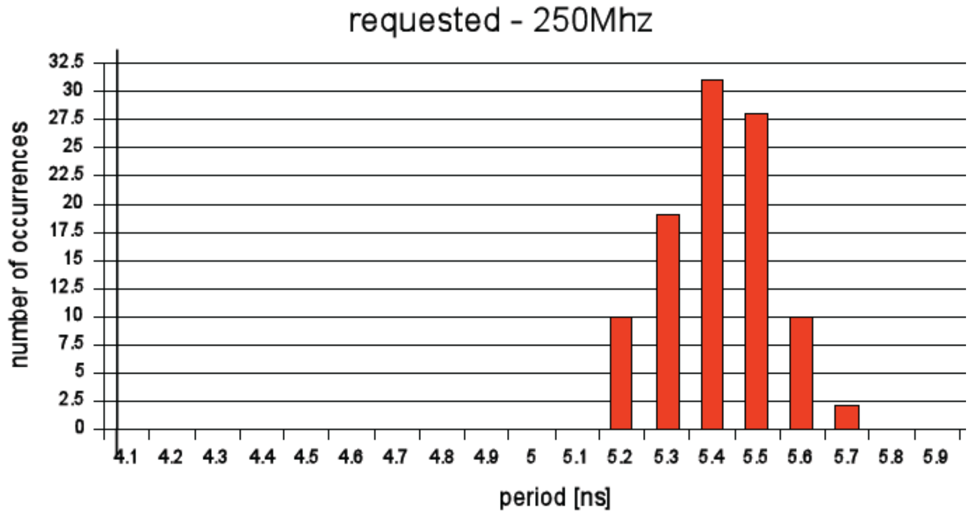
Multi-Pass Place-and-Route Analysis

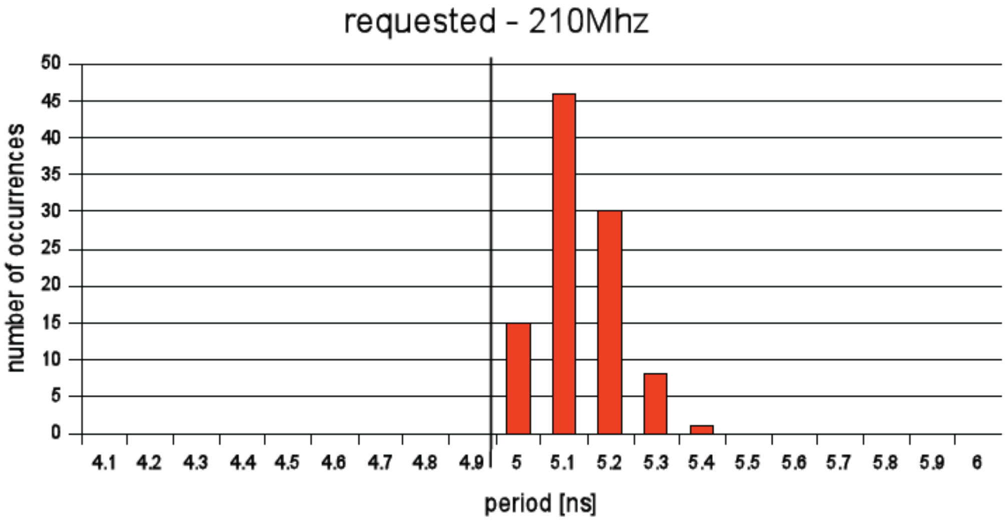
GMU SHA-512, Xilinx Virtex 5

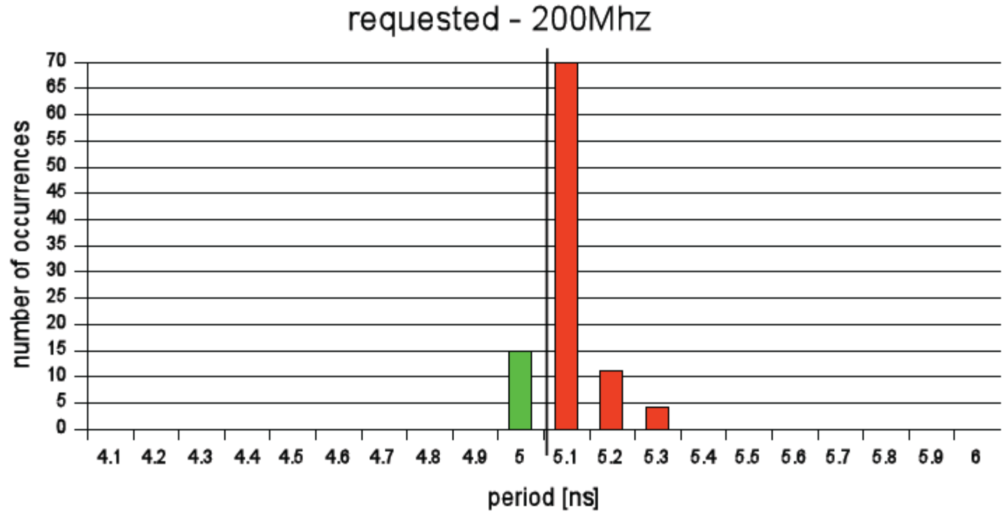
100 runs for different placement starting points

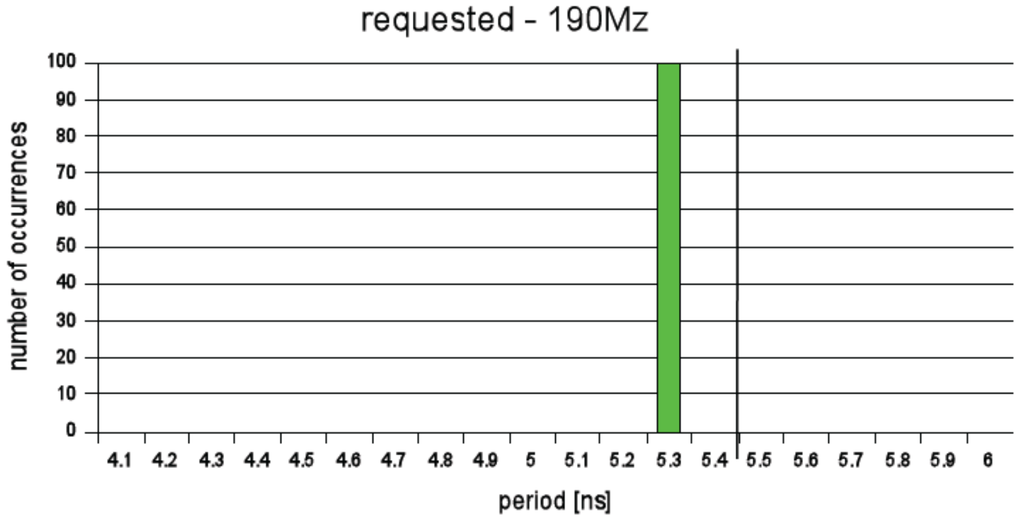


Dependence of Results on Requested Clock Frequency


requested - 250Mhz


requested - 210Mhz


requested - 200Mhz


requested - 190Mz

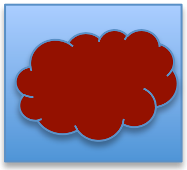
My Favorite Hardware Performance Metrics:

Mbit/s	**for**	**Throughput**
ns	**for**	**Latency**

Allows for easy cross-comparison among implementations in software (microprocessors), FPGAs (various vendors), ASICs (various libraries)

How to measure hardware cost in FPGAs?

1. Stand-alone cryptographic core on FPGA



Cost of a smallest FPGA that can fit the core.
Unit: USD [FPGA vendors would need to publish MSRP (manufacturer’s suggested retail price) of their chips] – not very likely
or **size of the chip in mm^2** - easy to obtain

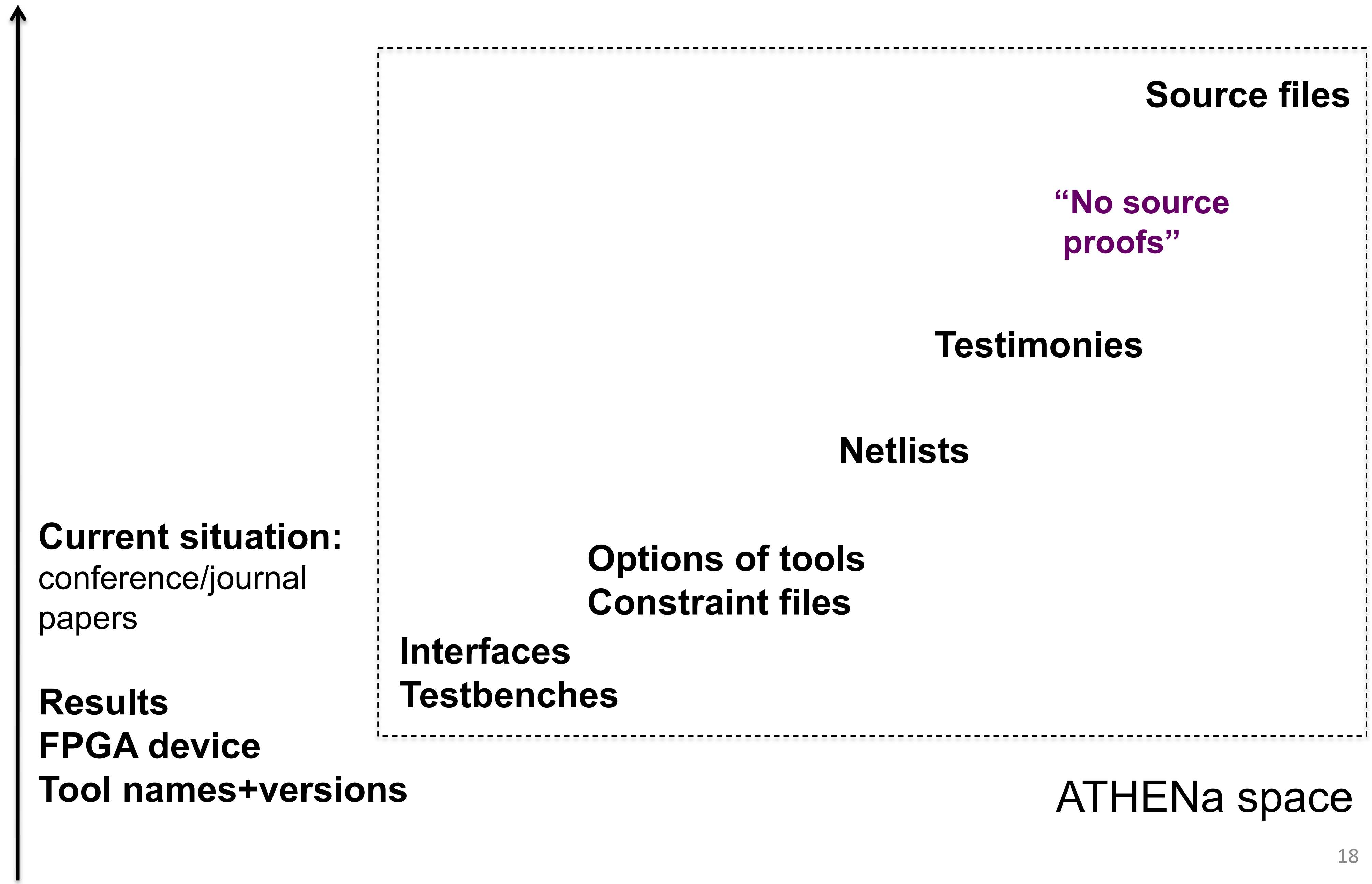
2. Part of an FPGA System On-Chip

Vector: **(CLB slices, BRAMs, MULs, DSP units)** for Xilinx
(LEs, memory bits, PLLs, MULs, DSP units) for Altera

3. FPGA prototype of an ASIC implementation

Force the implementation using only reconfigurable logic (no DSPs or multipliers, distributed memory vs. BRAM):
Use **CLB slices** as a metric. [LEs for Altera]

Level of openness



Current situation:
conference/journal papers

Results
FPGA device
Tool names+versions

Source files

"No source proofs"

Testimonies

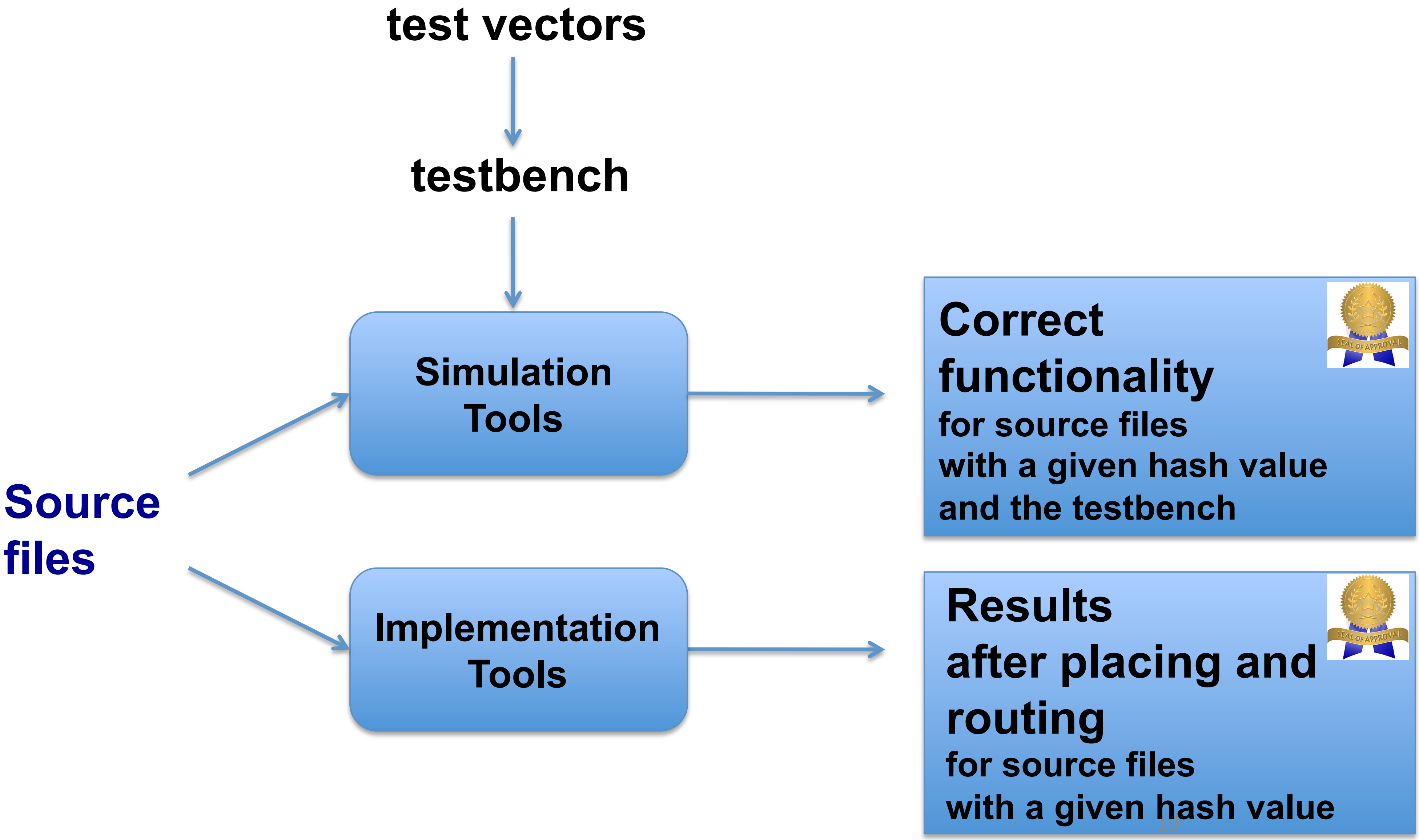
Netlists

Options of tools
Constraint files

Interfaces
Testbenches

ATHENa space

No Source Proof



test vectors

testbench

Simulation Tools

Implementation Tools

Source files

Correct functionality
for source files
with a given hash value
and the testbench

Results after placing and routing
for source files
with a given hash value