

EE380: Conflict and Technology

Oskar Mencer

September 25, 2019



1982

```
div8:
    ld c, 0
    or a
div8_lp:
    sub b
    jp c, no_div8
    inc c
    jp div8_lp
no_div8:
    add b
    ret
```





Computer Systems Laboratory Colloquium

4:15PM, Wednesday, May 2nd, 2001
NEC Auditorium, Gates Computer Science Building B03

EE380
2001

Computing with FPGAs

Oskar Mencer
Lucent / Bell Labs, and Imperial College, London

About the talk:

Field-Programmable Gate Arrays (FPGAs) can outperform microprocessors on certain tasks by many orders of magnitude. The open research problems of computing with FPGAs are: (1) understanding the limitations of FPGAs when competing with microprocessors, and (2) providing a useful programming methodology.

First, I will show how FPGAs can be utilized to accelerate certain algorithms by up to three orders of magnitude. Examples for methods achieving the speedups are:

- a. exploring parallelism and pipelining on the bit-level,
- b. optimizing the encoding of data values (number representation), and/or
- c. reducing the required memory bandwidth by implementing data-structures and algorithms directly on the FPGA.

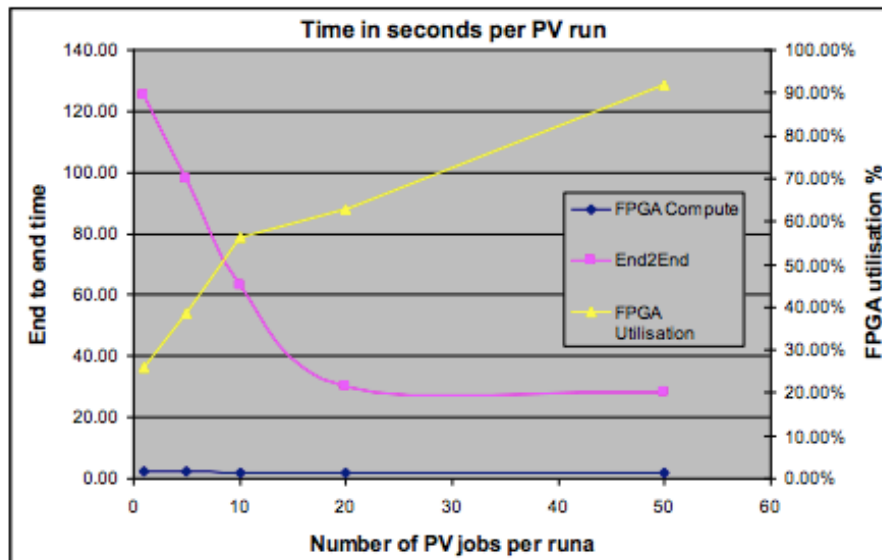
In addition, the speedup could be translated into savings in power consumption.

Second, I suggest a programming methodology for FPGAs based on Domain Specific Compilers. Domain specific compilers implement a divide-and-conquer, bottom-up approach to programming FPGAs. The vast space of possible architectures fragments into architecture families, which indirectly defines application domains. A domain specific compiler targets one architecture family and thus focuses on a single application domain. The StReAm compiler, under development at Bell Labs and Imperial College targets pipelined data-flow graphs mapped directly from object-oriented C++ to hardware. The goal is to provide a simple abstraction for programming FPGAs analogous to the abstraction of a microprocessor provided by the C programming language.

Technology in banking – a problem in scale and complexity

EE380
JP Morgan
2011

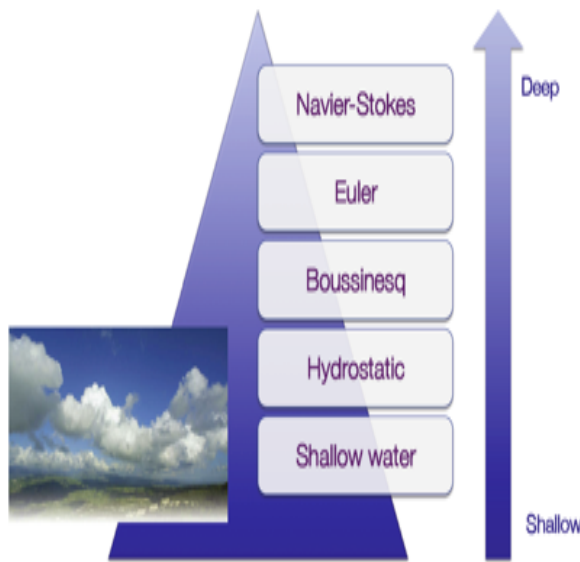
Stanford University, 11 May 2011



JP Morgan has taken a 20% stake in Maxeler – a key example of its commitment of using innovation to achieve strategic competitive advantage

Number of Scenarios	FPGA Compute	End2End	FPGA Utilisation
1	2.57	125.21	25.99%
5	2.35	98.02	38.54%
10	2.06	66.68	56.24%
20	1.86	30.88	62.63%
50	1.80	28.27	91.97%

Building the fastest programmable computers in the world



[L. Gan, et al, “Accelerating solvers for global atmospheric equations,” 2013]

Platform	<u>Performance</u>	<u>Speedup</u>	<u>Efficiency</u>	<u>Energy Improvement</u>
6-core CPU	4.66K	1	20.71	1
Tianhe-1A node	110.38K	23x	306.6	14.8x
MaxWorkstation	468.1K	100x	2.52K	121.6x
Maxeler MPC-X	1.54M	330x	3K	144.9x

14x

9x

Fastest computers for top 4 HPC applications

Applications	small case		big case		
	TTS [sec]	ETS [kWh]	TTS [sec]	ETS [kWh]	
BQCD	105	0.44	1,704.00	4.3	9x
BQCD (Ref: 1 rack Blue Gene/Q)			2,311.00	37.5	
NEMO (DFE)	388	0.164	1,945.00	3.8	7x
NEMO (Ref: 8,192 cores Cray XC30)			1,942.00	28.0	
QUANTUM ESPRESSO	32	0.013	3,210.00	7.6	5x
QUANTUM ESPRESSO (Ref: 2 racks BlueGene/Q)			3,200.00	36.4	
SPECFEM3D	232	0.096	5,150.00	77.2	4x
SPECFEM3D (Ref: 468 nodes Curie cluster)			15,500.00	297.5	

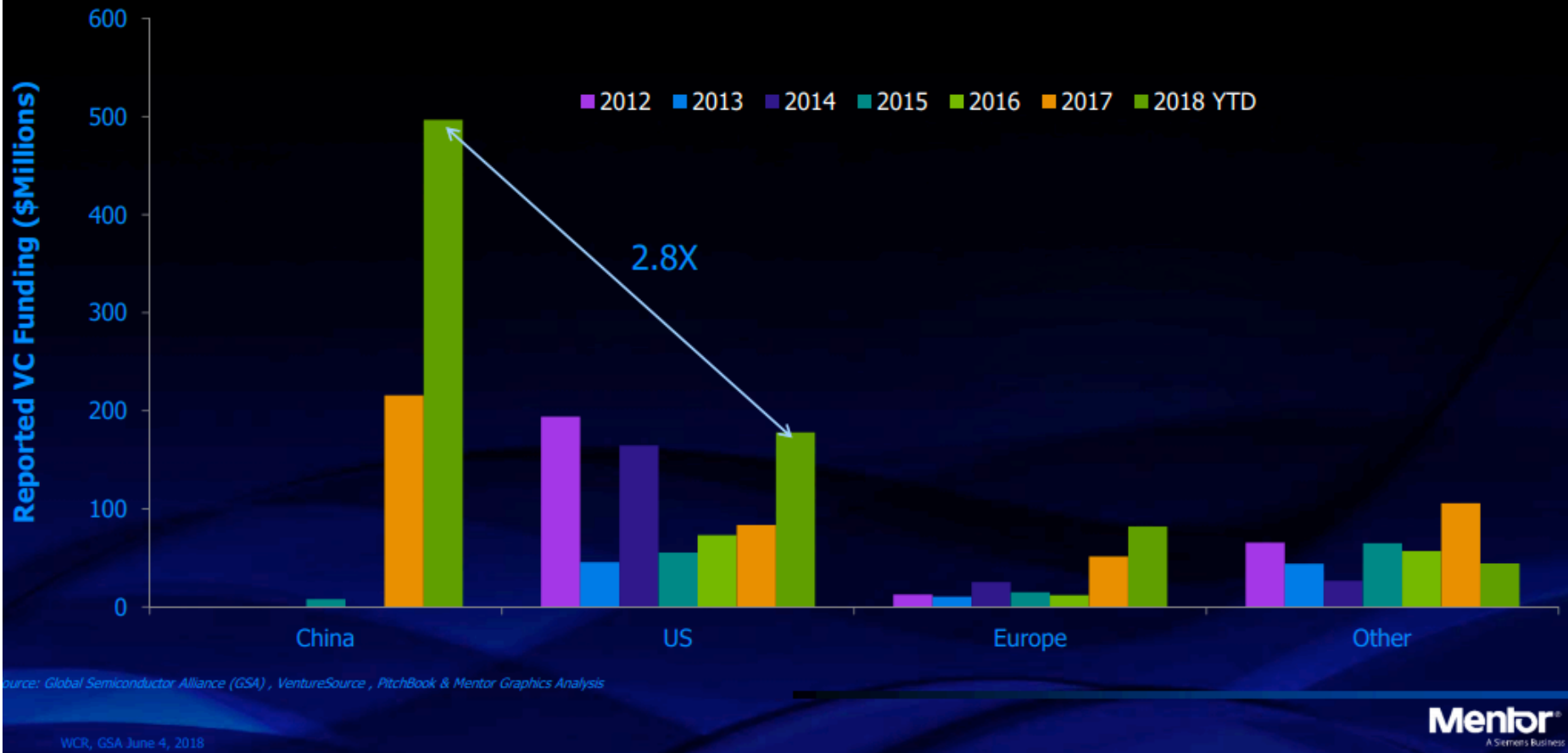
“Commercializing FPGAs for Computation”

Xilinx Share Price



China vs US funding for Fabless Semiconductor

China 2.8X US in 2018 YTD Funding Worldwide Fabless Company Venture Capital Rounds 1-3



courtesy of Wally Rhines, Mentor Graphics, now Siemens

“[Force] persisted through a series of conflicts, then vanished of itself---what's the expression---ah, yes, 'not with a bang, but a whimper,' as the economic and social environment changed. And then, new problems, and a new series of wars.”

Isaac Asimov, I, Robot, quoting T S Elliot (thanks to Dennis Allison)

Conflicts

We live in a world of many conflicts:

Conflict between US and China

Conflict between CPUs, GPUs and FPGAs

Conflict between VHDL and HLS people

Conflict between SW people and HW people

Conflict between Internal IT and Small Suppliers

Conflict between Bank Traders and Operations

Conflict between Employees and Management

Conflict between small and LARGE companies

NIH, Change due to new product, inertia

Conflict between old Conflicts and new Conflicts

New Conflicts

Populism vs Anti-populism

The Internet vs Democracy

Quantum Computing vs Nay sayers

Global Warming vs Mars Explorers

Observation 1:

Thermodynamics says Entropy increases or stays the same, similarly, from an individual perspective, the number of conflicts we participate in, seems to increase as time progresses.

Observation 2:

Energy conservation plus increase in # of conflicts means that personal Energy (and time) per conflict is going down.

The Kill Switch Product Idea

Knight Capital Group



The Knight Capital Group was an American global financial services firm engaging in market making, electronic execution, and institutional sales and trading. With its high-frequency trading algorithms Knight was the largest trader in U.S. equities, with a market share of 17.3% on NYSE and 16.9% on NASDAQ. [Wikipedia](#)

Type: Subsidiary

my calendar entry

Traded as: NYSE: [], until July 1, 2013

Industry: Financial Services

■ Knight Capital visit - Amit, Doug

Wednesday, February 1, 2012 · 9:30 – 11:00pm



545 Washington Boulevard, Jersey City, New Jersey 07

Conflict in the real world



Same pictures with a different perspective:

What happened after these two pictures were taken?

Uncertainty

How do we distinguish news from fake news

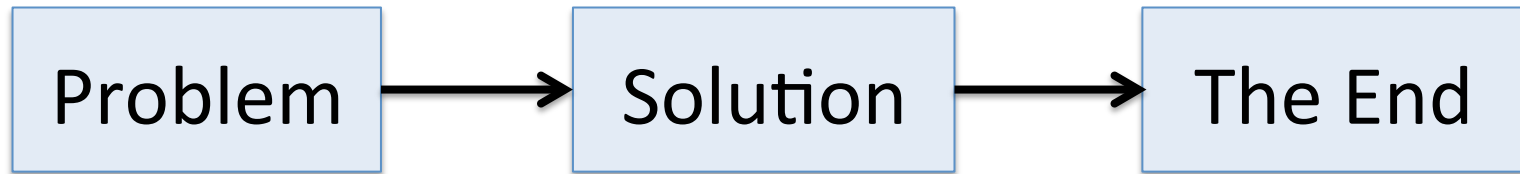
Chip Wars, Market Forces, and Disruptive Tech

Can AI predict which company will be around a year from now?

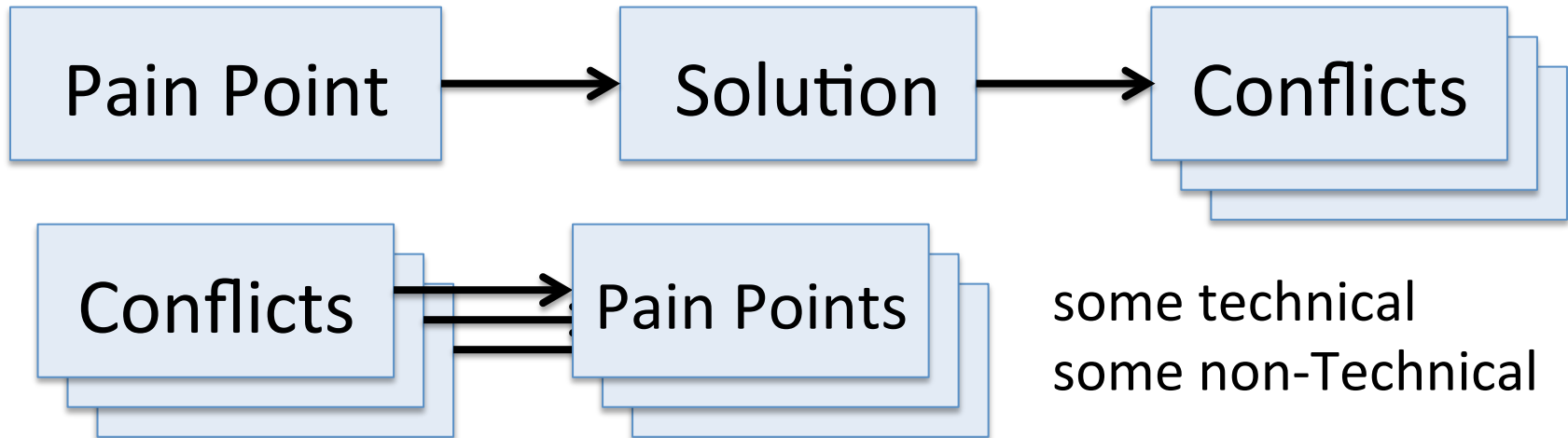
Observation 3 (follows from “Efficient Markets Theory”)

With computers (AI) predicting the future,
the future is getting more and more unpredictable.

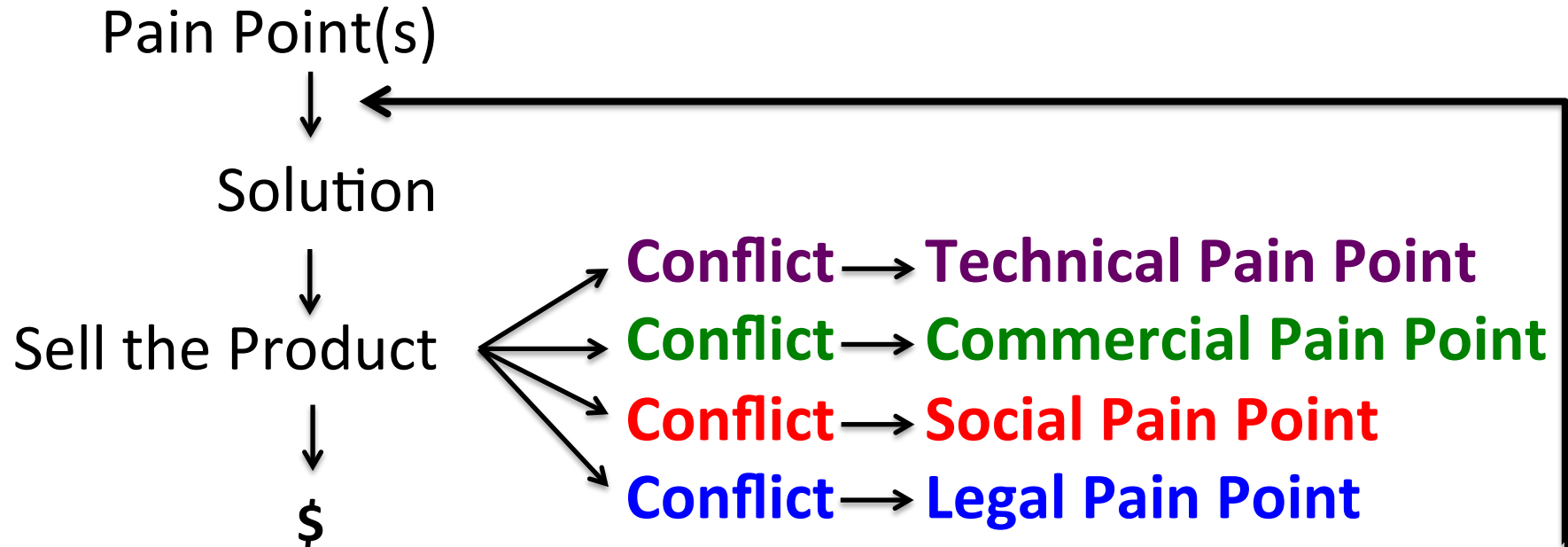
The Homework Problem



The Real World



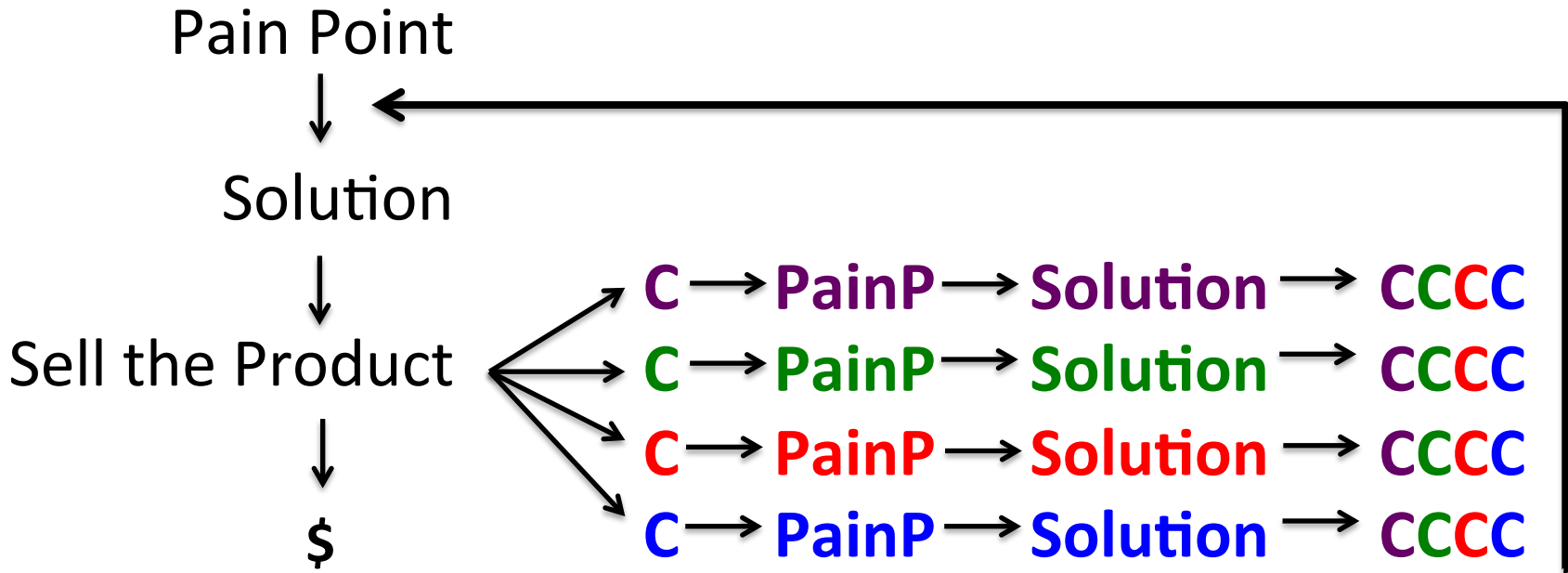
Start a company, build a product



Product Plan:

1. Identify the pain point solved by your product
2. Identify the conflicts caused by your product
3. Identify the new pain points and solutions or sell a product and see what happens.

Scaling is a race against cashflow



It's a state machine with the state being the cash in the bank.
Scaling success is then a function of speed of resolving conflicts.

Top 10 Conflicts in Computing with FPGAs

Conflict 1: HDL is hard, need a high level programming language

**Conflict 2: FPGAs DRAM Memory interfaces
are slower than CPU and GPU**

**Conflict 3: FPGA floating point is not IEEE compliant and inefficient
(due to the barrel shifter)**

Conflict 4: Separating CPUs and FPGAs threatens CPU vendors

Conflict 5: There are no applications for FPGAs

**Conflict 6: Need to rewrite
parts of the application**

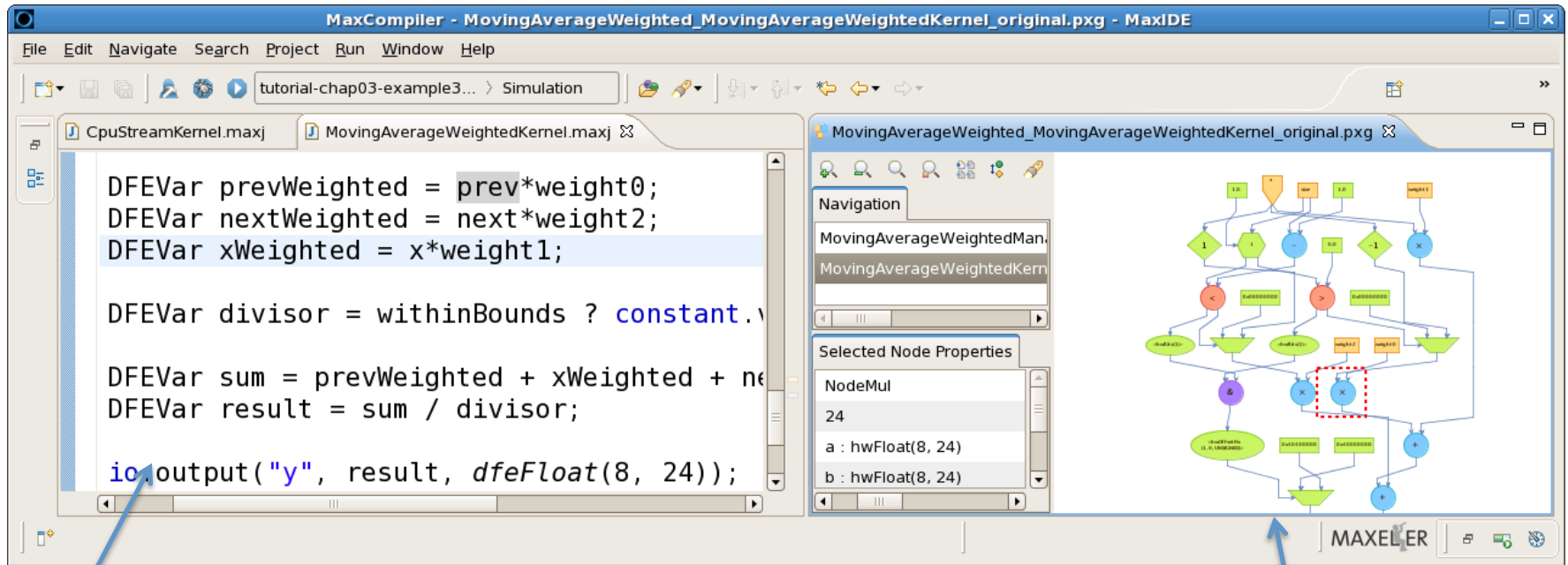
Conflict 7: Debugging hardware is hard

Conflict 8: Place-and-Route takes 3 days

Conflict 9: A high level language obsoletes the HDL experts

Conflict 10: Most software does not need (hardware) acceleration

C1: HDL is hard, high level programming



Dataflow Simulator 100x faster than VHDL simulation

MaxJ Language embedded in Java

Corresponding Dataflow Graph

C1: Connect language to space on the chip

The goal is to maximize utilization of resources on the chip, **and bandwidth on the memory bus.**

LUTs	FFs	BRAMs	DSPs	
727	871	1.0	2	: MyKernel.java
				: resources used by this file
0.24%	0.15%	0.09%	0.10%	: % of available
71.41%	61.82%	100.00%	100.00%	: % of total used
94.29%	97.21%	100.00%	100.00%	: % of user resources
				:
				: public class MyKernel extends Kernel {
				: public MyKernel (KernelParameters parameters) {
				: super(parameters);
1	31	0.0	0	: DFEVar p = io.input("p", dfeFloat(8,24));
2	9	0.0	0	: DFEVar q = io.input("q", dfeUInt(8));
				: DFEVar offset = io.scalarInput("offset");
8	8	0.0	0	: DFEVar addr = offset + q;
18	40	1.0	0	: DFEVar v = mem.romMapped("table", addr,
				: dfeFloat(8,24), 256);
139	145	0.0	2	: p = p * p;
401	541	0.0	0	: p = p + v;
				: io.output("r", p, dfeFloat(8,24));
				: }
				: }

C2: FPGAs DRAM Memory interfaces are slower than CPU and GPU

Solution 1: Use on-chip MB SRAM with >10TB/s access bandwidth

Maxeler tools help to restructure code to use SRAM

Solution 2: Put more DRAM on the FPGA card than the GPU

Maxeler cards with 96GB of DRAM when GPUs had 8GB

Solution 3: Build an FPGA with GDDR6

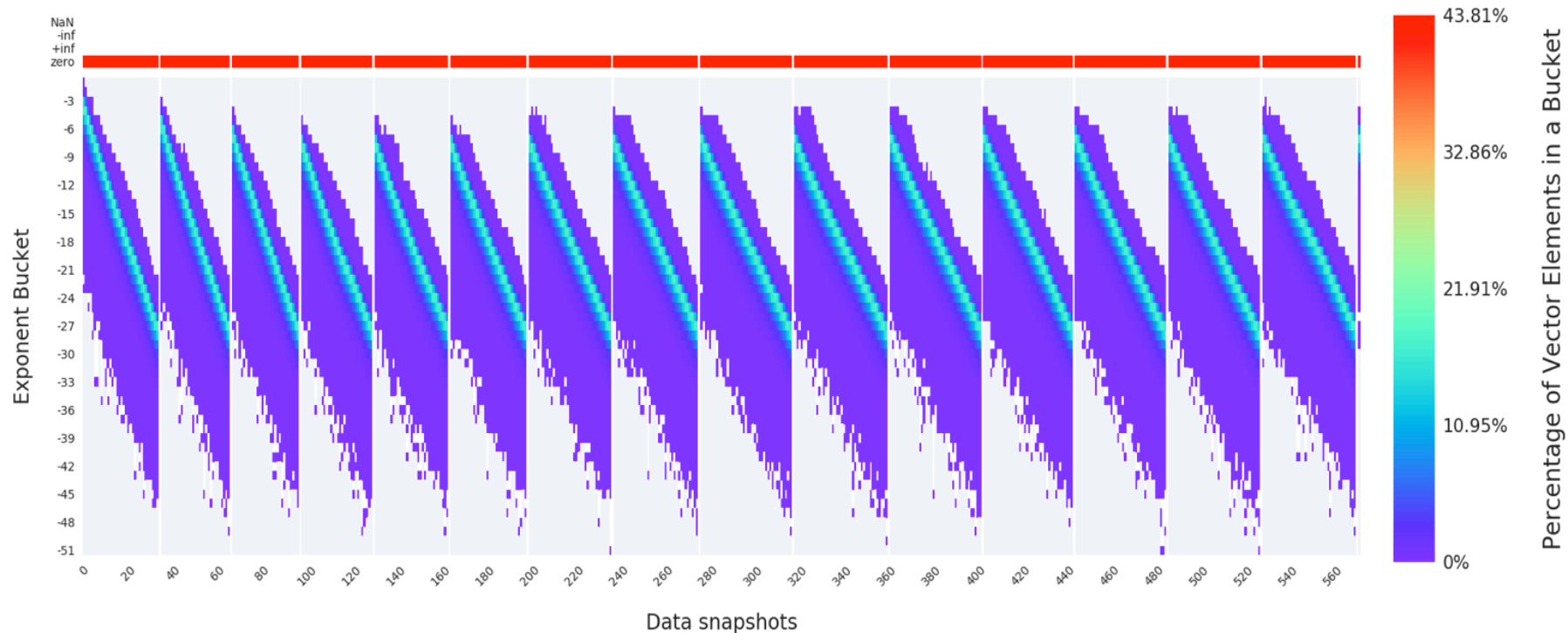
see new Achronix FPGA with GDDR6

Solution 4: Build an FPGA package with HBM,

see latest Xilinx VU31-47P with up to 16GB of HBM

C3: FPGA floating point is inefficient (due to the barrel shifter)

Evolution of 'r' values through iterations



Maxeler Numerics Analysis and Visualization Tool

C4: Separating CPUs and FPGAs

Conflict: CPU and FPGA in the same server is inefficient.
The optimal balance between FPGAs and CPUs
is never exactly 50-50, Server+FPGA card is inefficient

Solution: build an Infiniband-connected appliance

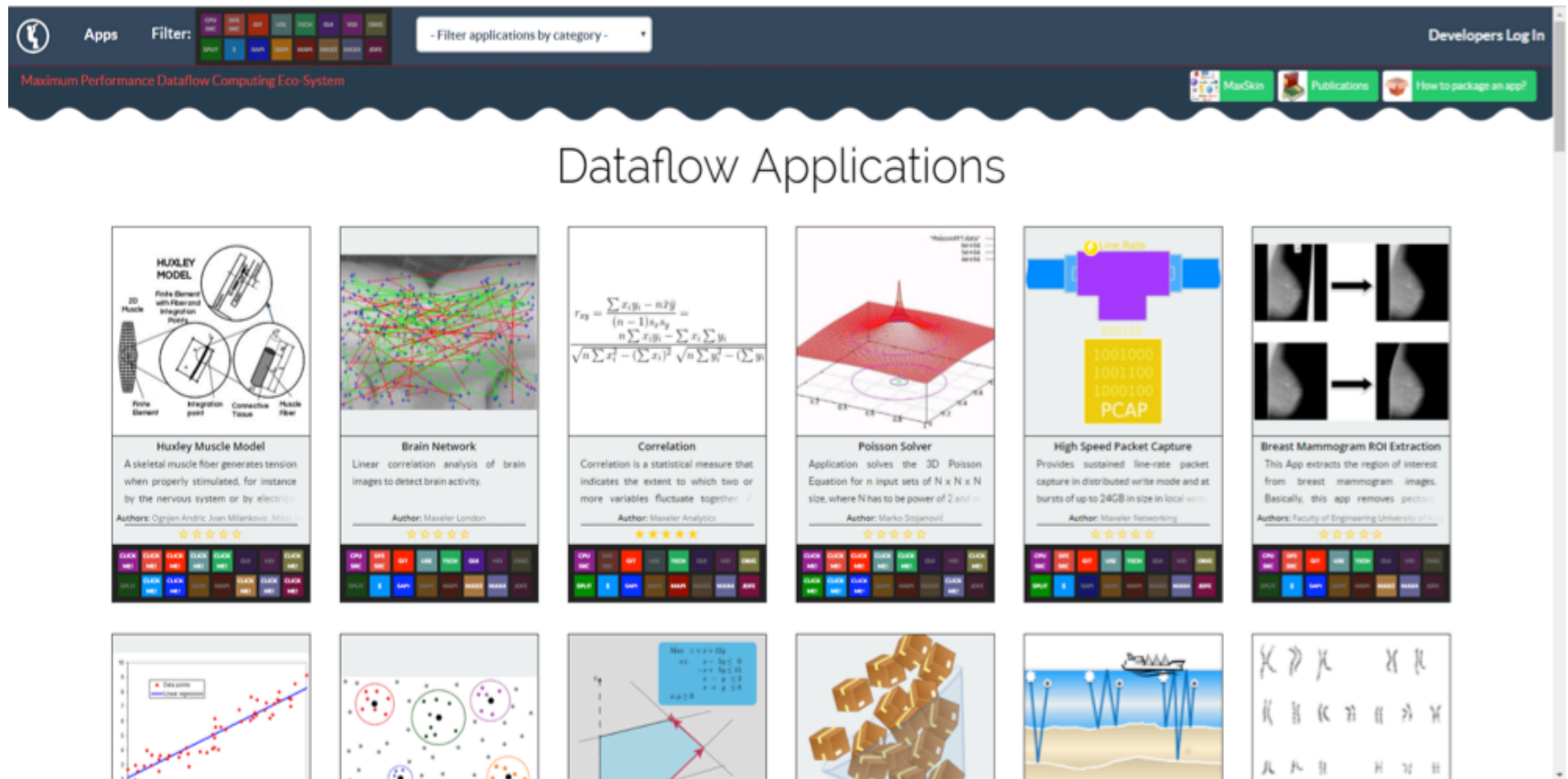


**New Conflict: Server vendors see the FPGA appliance
as a threat, stealing computation away from the CPU.**

New Conflict: Infiniband was banned in Bank datacenters

C5: There are no Applications for FPGAs

Why would you buy a computer for which there are no applications



<http://appgallery.maxeler.com/>

C6: Need to rewrite parts of the application

Solution 1: Develop the Maxeler acceleration process

New Conflict: We are changing the code, maintained by a software expert, making it compile only with our proprietary tool, on our proprietary hardware!

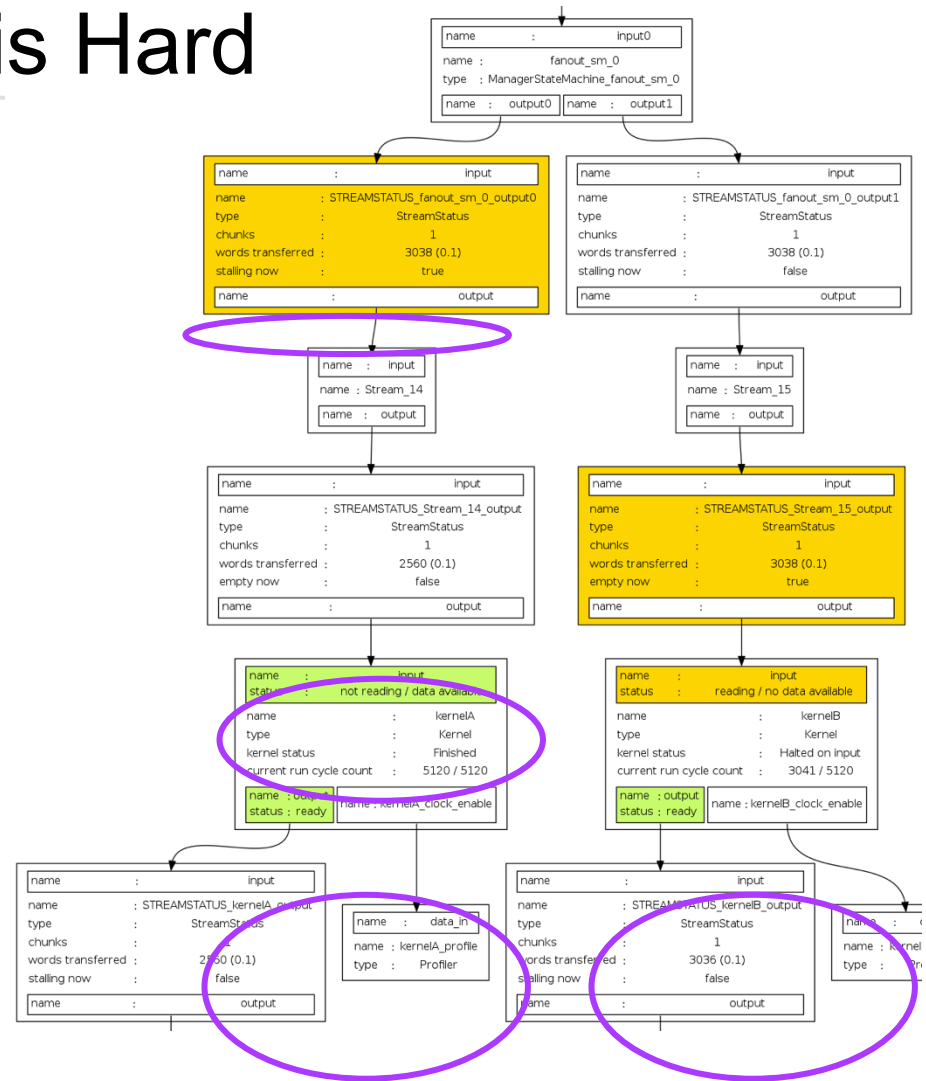
Solution 2: nVidia convinced the world that it is ok to rewrite parts of the software source code with CUDA.

Solution 3: BigStream, the VM of acceleration for
Kafka, Tensorflow, Spark

C7: Hardware Debug is Hard

MaxDebug tool example

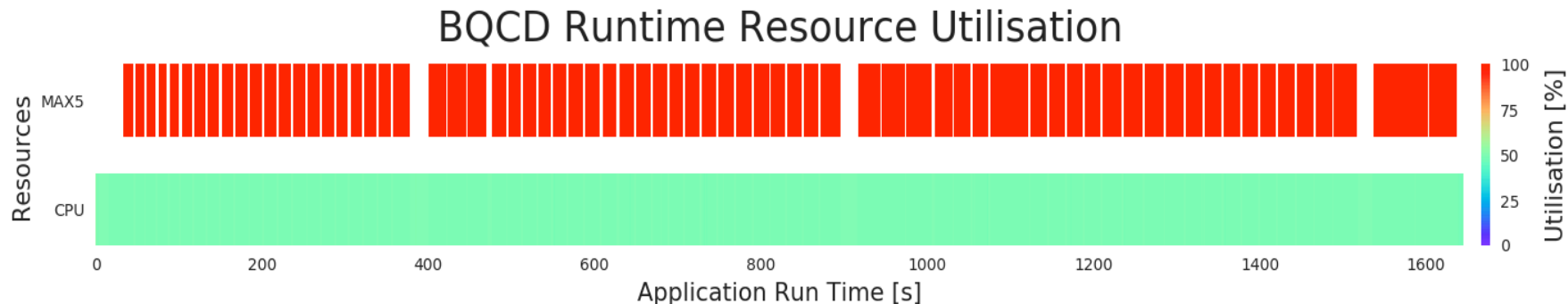
- 3038 words transferred into the input buffer of kernelA
- 2560 words transferred from that buffer into kernelA
- kernelA has finished all its ticks
- 2560 words transferred out of kernelA
- Meanwhile kernelB is not done and is waiting for more data



Conclusion: KernelA has not been assigned the correct number of ticks!

C7' Hardware Efficiency Debug is Hard

Maxeler Dynamic Dataflow Event Viewer
extracting parallelism and monitoring efficiency



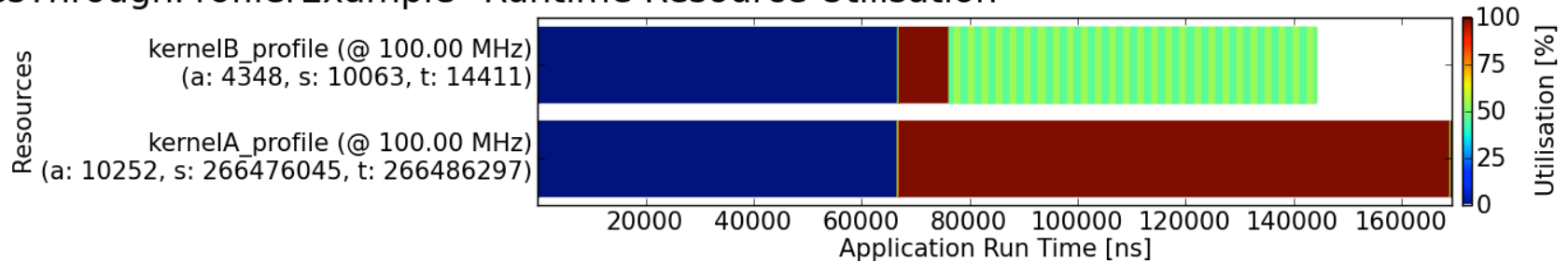
Shows dataflow balance between processing units

Balancing execution is hard work!

C7” Hardware Performance Debug is Hard

MaxProfile tool example

"PassThroughProfilerExample" Runtime Resource Utilisation



kernelA and kernelB both receive data from same src

kernelA consumes (and produces) data more slowly

kernelB's utilisation hovers around 50%

- kernelB has to wait for more data, because:
- Upstream the pipeline is stalled
- because kernelA does not consume fast enough
- Remedies: more pipes in kernelA, increase clock A

C8: Place-and-route takes 3 days

Solution 1: Build a Place&Route cluster and a Place&Route job distribution system (MaxQ)

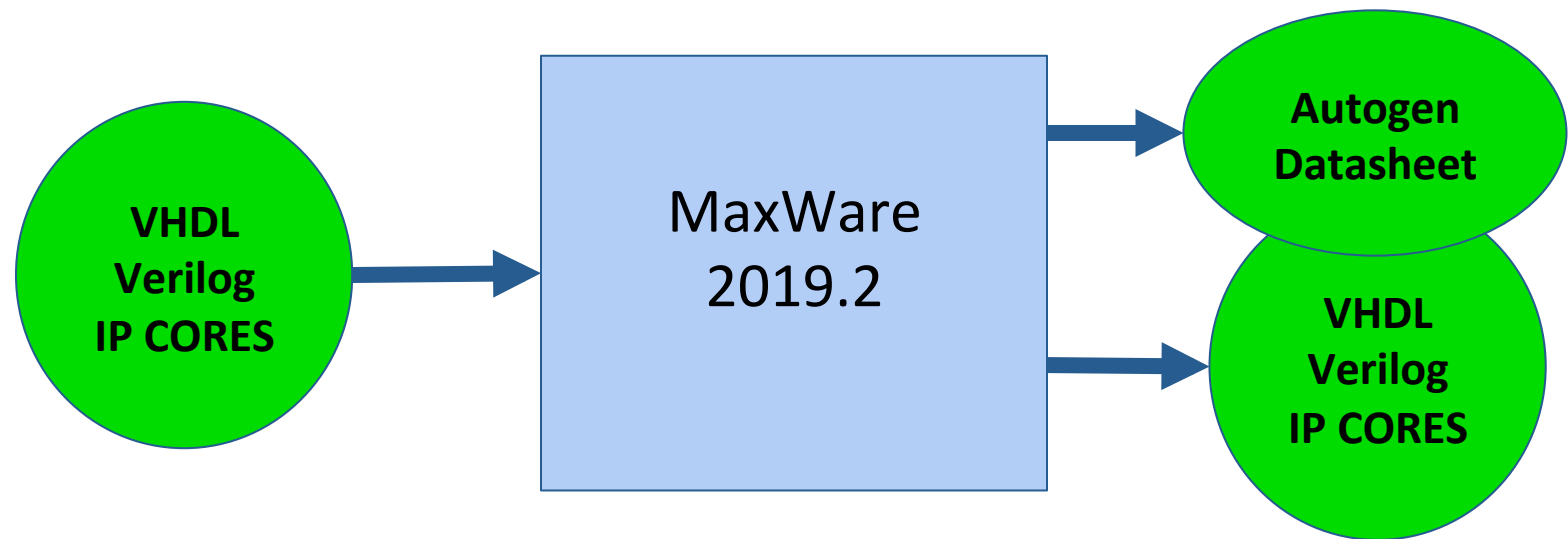
Solution 2: Ask Xilinx and Altera to let us accelerate Place&Route on FPGAs

New Conflict: Internal Software teams regard the Place&Route software as key competitive differentiator

Solution 3: Make architectural changes to the FPGA and restrict circuit types on high level to reduce Place&Route time.

C9: High level language obsoletes the HDL expert

Solution: Change MaxJ to an HDL IP Core generation tool
(and allow import of 3rd party IP cores)



see www.maxeler.com/ip-cores.html

C10: Most software does not need acceleration

SQL

```
s = run_sql_procedure('uses_cpu_per_impac_server_mys
```

Running SQL:

Duration: 114.239367 seconds

Python

```
p = 
```

Running Python:

Duration: 0.952739 seconds

```
speedup(p, s)
```

Speedup: 119.9x



120x faster and no new hardware is needed!

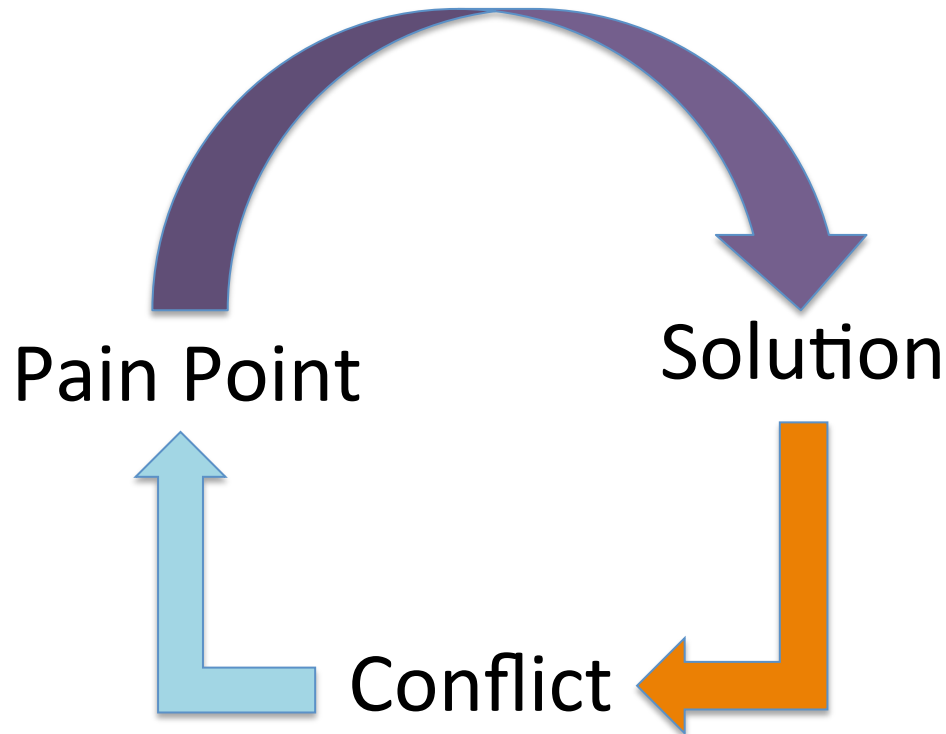
Top 2nd Generation Conflicts in Computing with FPGAs

- Conflict 1:** If 1 rack of FPGAs replaces 10 racks of CPUs, the CPU vendors sell 10x less hardware
- Conflict 2:** If a CyberSecurity product with FPGAs replaces a \$1M w/ a \$100K solution, current vendor loses 10x revenue
- Conflict 3:** If FPGAs accelerate computation by 10x, then data hits the networking infrastructure at 10x higher velocity
- Conflict 4:** If the FPGA solution means changing vendor, then stability of the supply chain may be in danger
- Conflict 5:** If computing with FPGA brings a new language, some people may not like the new language
- Conflict 6:** If FPGAs do not use the same arithmetic as processors, governments have to re-qualify regulatory computations

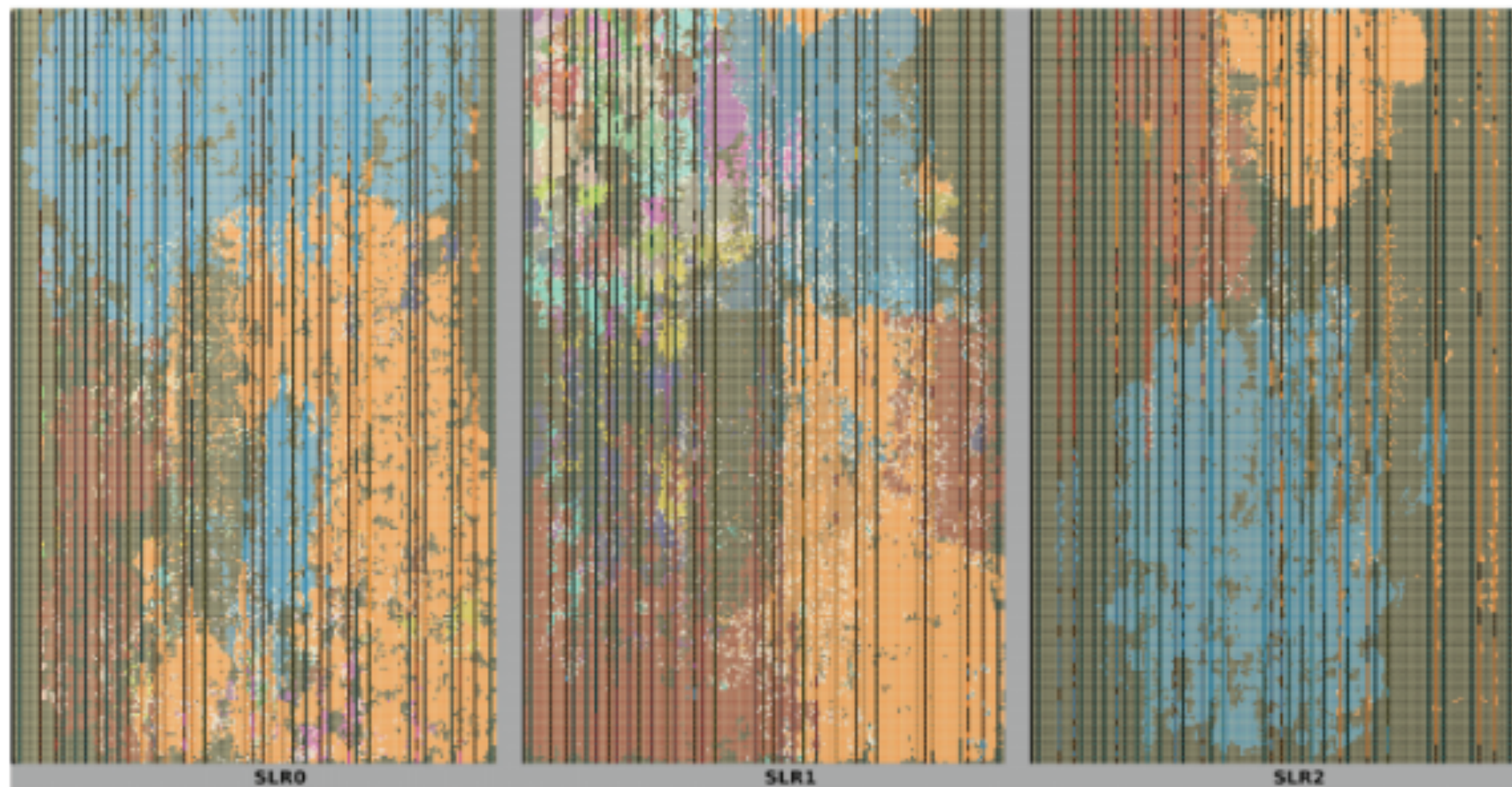
.....

Conclusions

To scale, you need to keep up with the conflict cycle, predict and solve the next next conflict before it happens!



World's hardest simulation, Quantum Chromodynamics on a Xilinx VU9P FPGA



Aerial view of the BQCD conjugate gradient bitstream. This particular design uses subdomains of size $16 \times 16 \times 16 \times (LT)$ and can simulate any grid size that is a multiple of this subdomain size