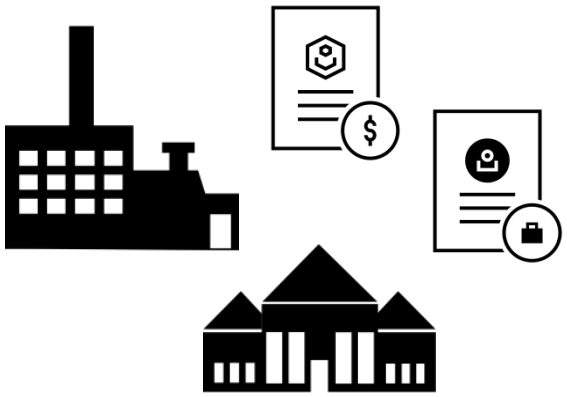
Hardware Accelerated Application Integration: Challenges and Opportunities

Active @ ACM/IFIP/USENIX Middleware 2017

Daniel Ritter

Application Integration



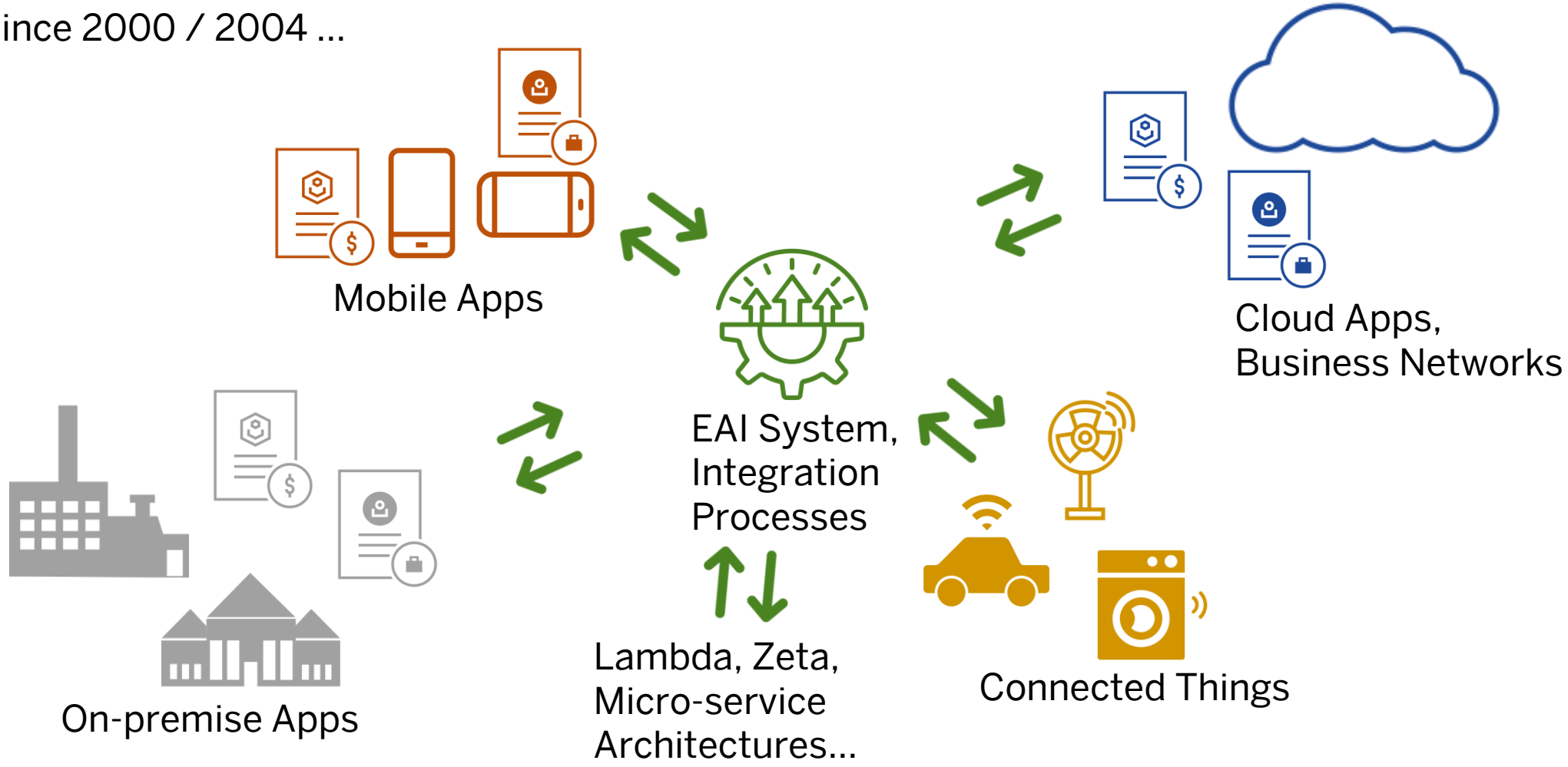
On-premise Apps

EAI System, Integration Processes

- De-coupling apps
- Solving n-square connection and variety problems (for textual data) [Lin2000]
- Message routing and transformation patterns from 2004 [HW2004]

Emerging Application Integration

since 2000 / 2004 ...



Emerging Trends lead to Challenges

On-premise Apps

Cloud Apps, Business Networks

Mobile Apps

Connected Things

Lambda, Zeta, Micro-service Architectures...

EAI System, Integration Processes

EAI Challenges:

- New variety problem, e.g., media message protocols
- Number of messages (Velocity)
- Message sizes (Volume)
- Fault-tolerance (Stability)
- ...

General Challenges:

- Data center efficiency
- Power consumption
- ...

Classical Solution Space

Challenges:

Media message protocols (variety problem)
...

Conversion, User Interaction, ...

Message sizes (Volume)

Number of messages (Velocity)
...

per operation

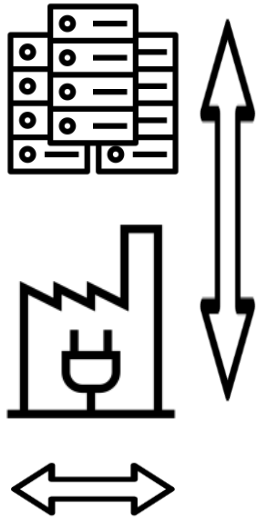
Fault-tolerance (Stability)
...

Solutions:

Scale (out, up), e.g., parallelization, batching

Software solutions, e.g., streaming, process / algebraic simplification, data reduction (up to sampling)
...

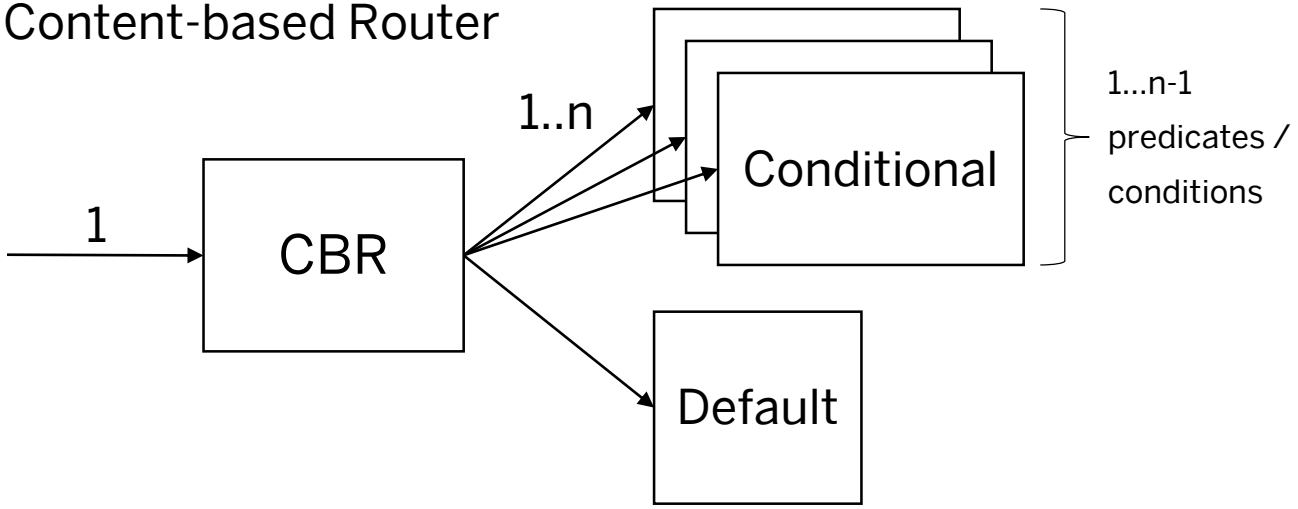
Side-effects:



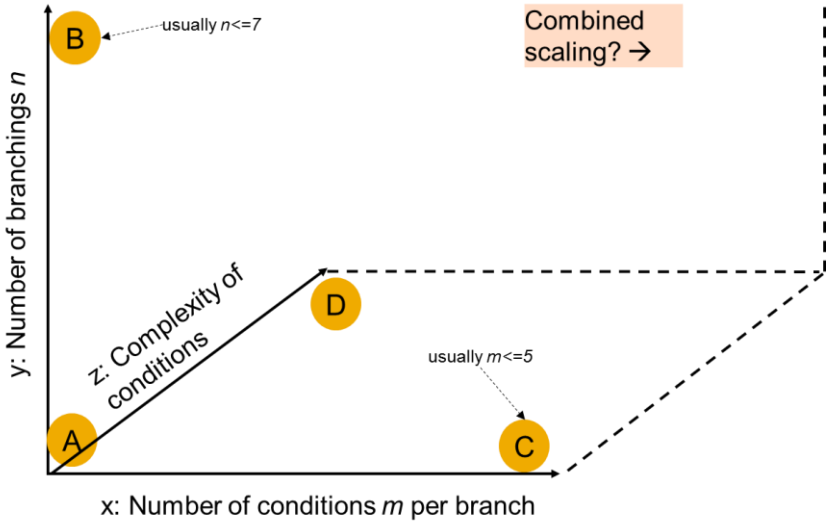
Hardware, Data center scaling, build power plants
...

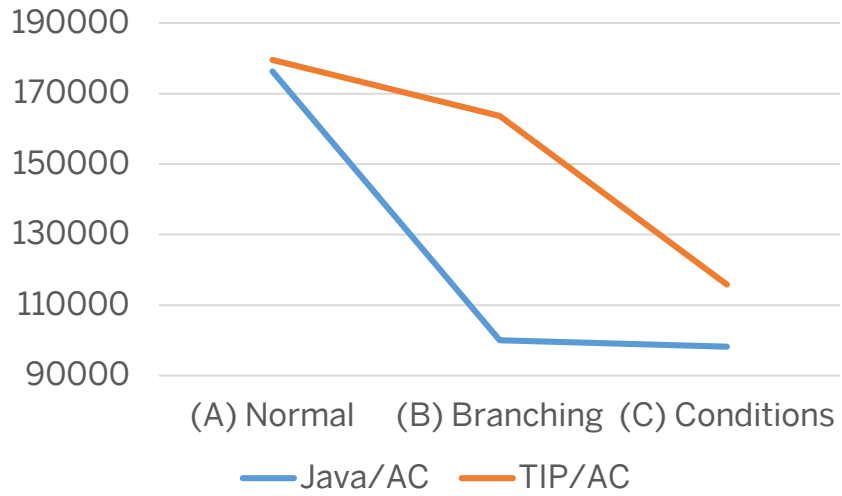
→ Software latency and throughput limited [LGMBEV2012]

Example: Message Routing



EIPBench Pattern Benchmark [RMSRM2016]; AC:=Apache Camel, TIP:=Vectorization



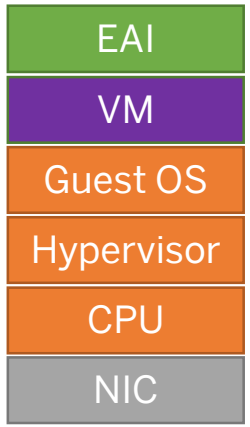


Hardware Acceleration

One step back, important EAI factors:

- Closeness to the network (e.g, connect two applications)
- Expressiveness (e.g., conditions, expressions)
- Efficiency (e.g., volume, velocity)
- Flexibility (e.g., change integration process)

Efficiency through Specialization



Current Software solutions

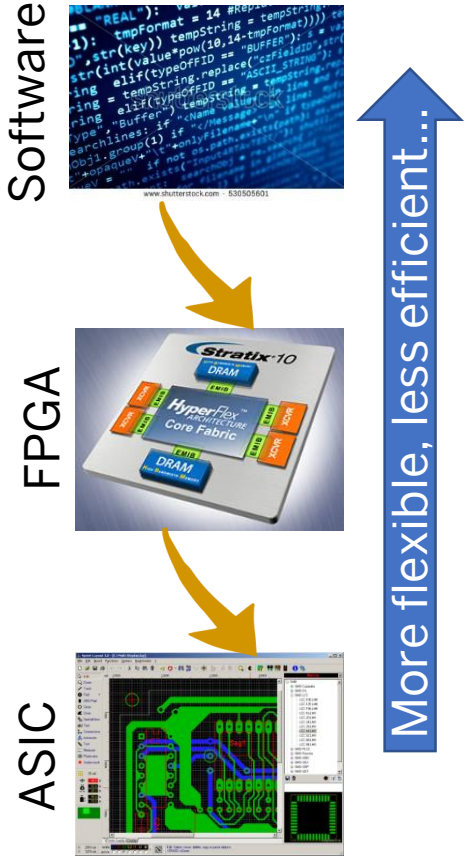
Classic SAAS

Why not use **FPGAs**?

Why not use **ASICs**, **GPUs** (+SIMD,−flops, power)?

Why not use **SDNs** (+network,−expressiveness: Integration patterns)?

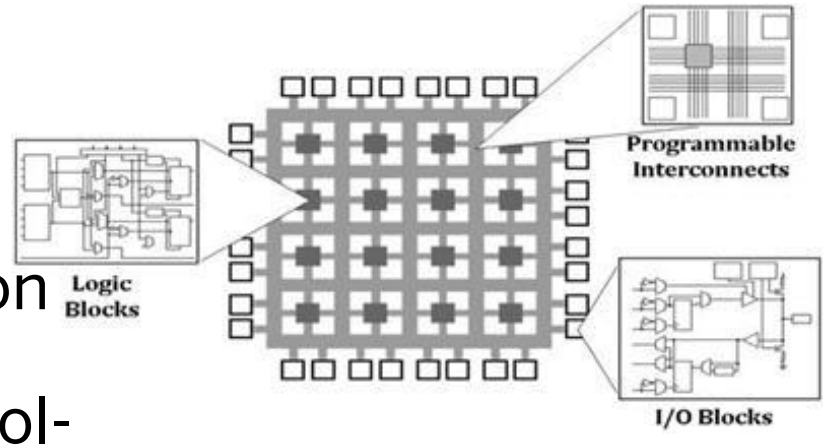
Use FPGAs due to good trade-off between flexibility vs efficiency; designs can become ASICs



similar to [Put2017]

What are FPGAs?

- Field Programmable Gate Arrays
- Fabric of interconnected logic blocks, on-chip memory, I/O
- Customize logic and I/O
- Reconfigurable hardware is more efficient than general-purpose hardware (CPU); reconfiguration times 100ms to 1s, partial reconfiguration
- FPGA ~ Dataflow architecture [C1986] vs. Control-flow architectures: single-, multi-core CPUs (von Neumann + beyond)

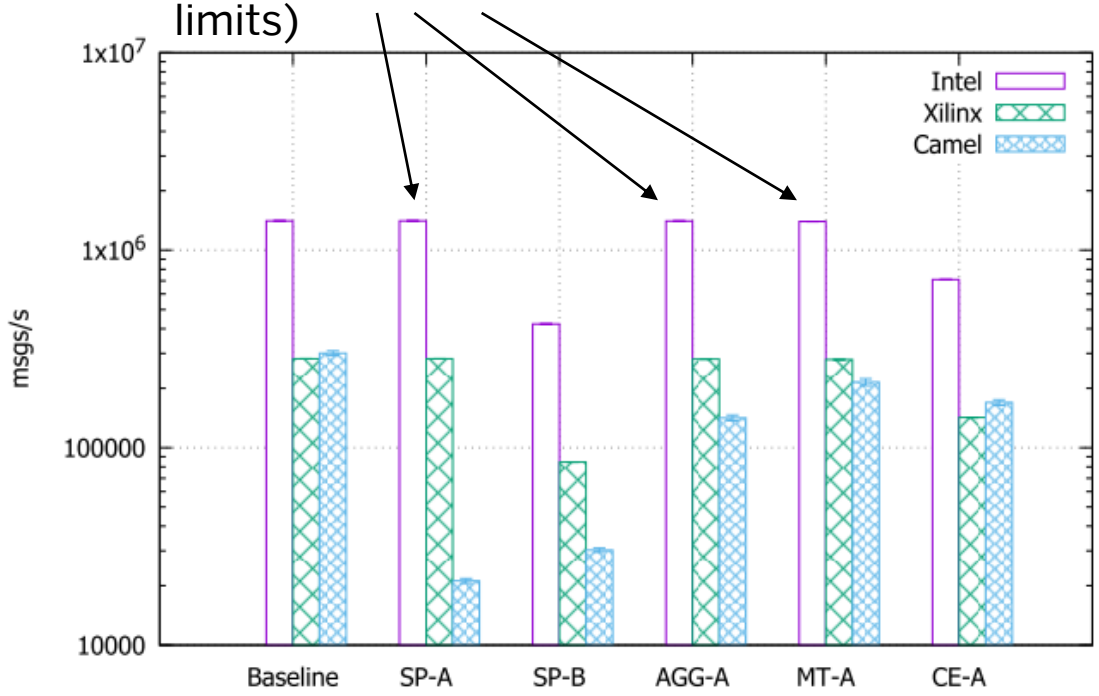


⊕ high degree of parallelism, streaming

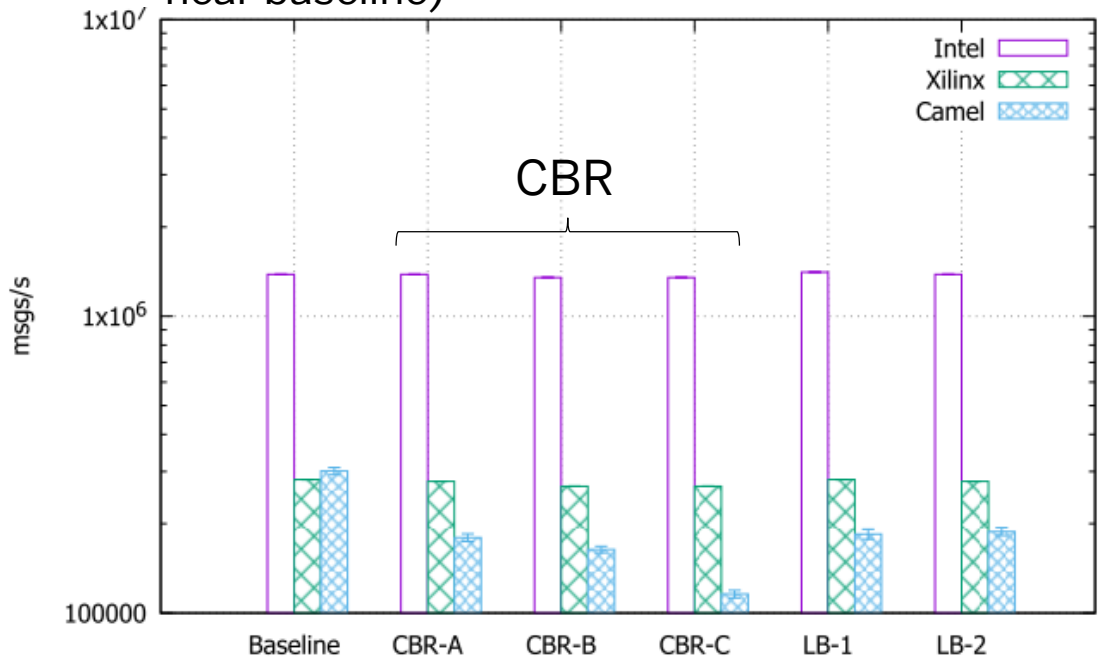
⊖ limited to resources on the chip

Message Throughput (revisited)

Even complex routing (e.g., SP, AGG) and transformation patterns (MT) have throughput close to baseline (i.e., hardware limits)



The throughput is invariant to multiple conditions and route branches (e.g., CBR-B+C, LB and join router (not shown) perform near baseline)



Disruptiveness ..

disruptive

[dis-**ruhp**-tiv]

Spell Syllables

Examples Word Origin

See more synonyms on Thesaurus.com

adjective

1. causing, tending to cause, or caused by disruption; disrupting :
   the disruptive effect of their rioting.
2. *Business.*
   a. relating to or noting a new product, service, or idea that radically changes an industry or business strategy, especially by creating a new market and disrupting an existing one:
      disruptive innovations such as the cell phone and the two-year community college.
   b. relating to or noting a business executive or company that introduces or is receptive to such innovation:
      disruptive CEOs with imagination and vision.

a. relating to or noting a new product, service, or idea that radically changes an industry or business strategy, especially by creating a new market and disrupting an existing one:

... in information systems requires:

(a) Novel types of applications

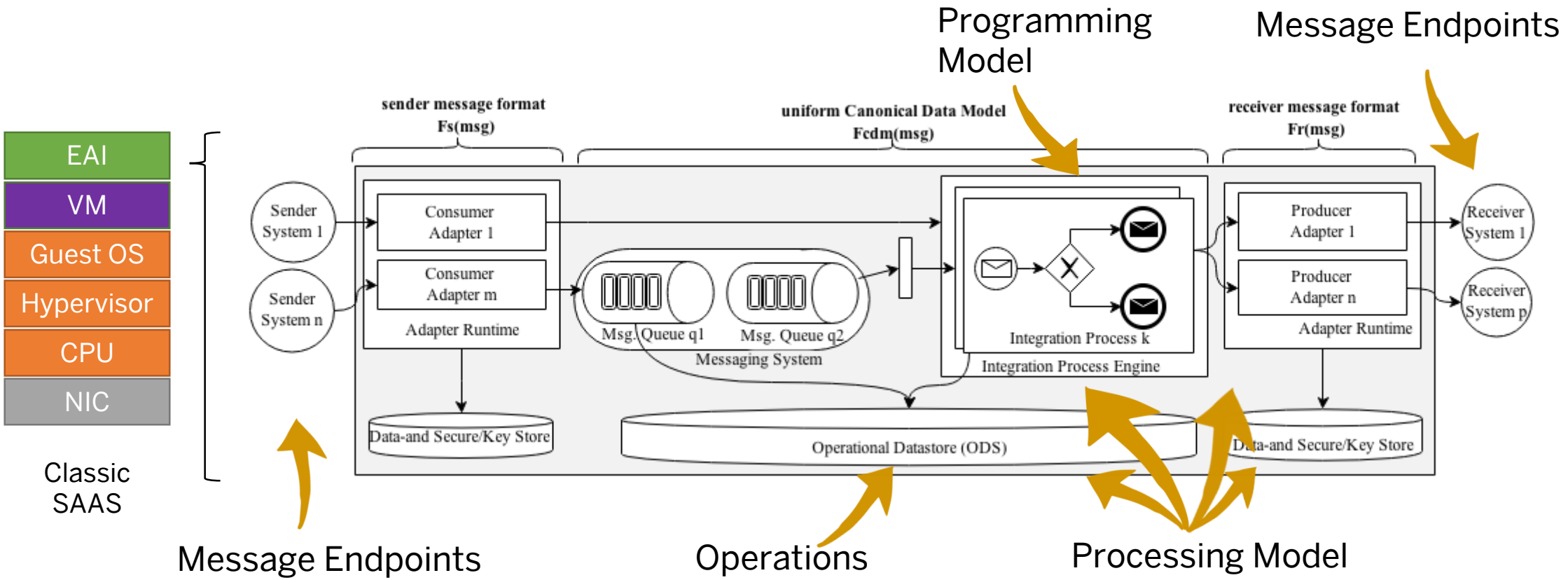
(b) Novel technology and hardware

Similar to Wolfgang Lehner's Keynote VLDB 2017

Crossroads of Middleware and Hardware

Challenges and Opportunities

EAI Architecture Aspects



Programming Models

Integration process modeling, configuration

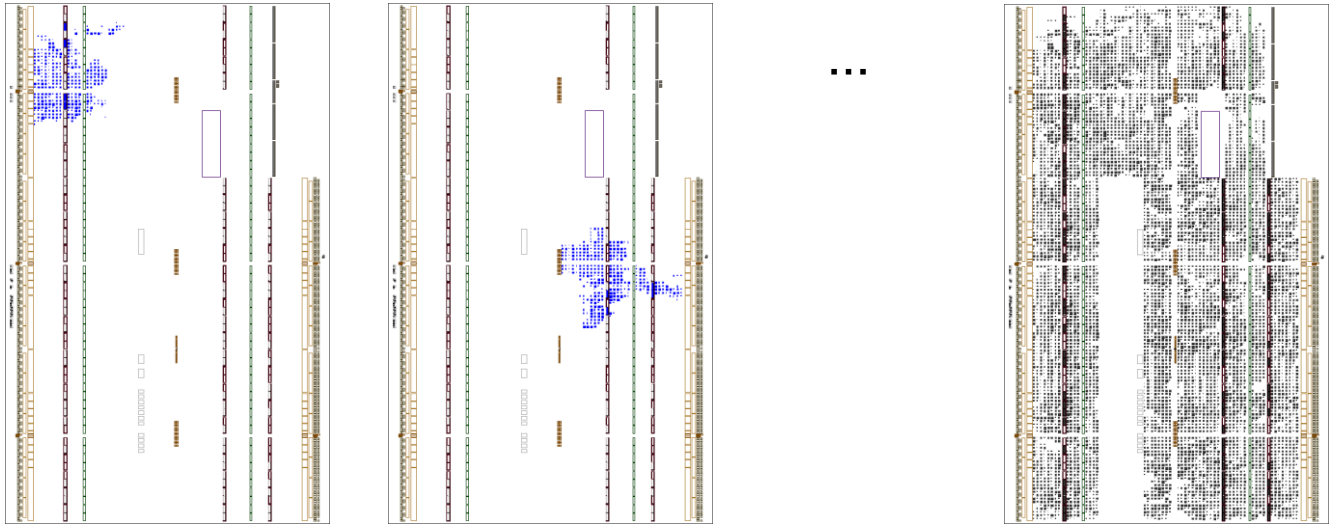
\+ FPGAs flexible, reconfigurable, became affordable

− FPGA development flow, lack the expertise to use the hardware-oriented FPGA, 10:1 or larger ratio of SW to HW programmers; UDFs space critical

Requires:

- Composable HDL / HW templates for building blocks (patterns) [RMRM2017])
- High-level synthesis of conditions / expressions (OpenCL,)
- Better editors and flow (PSHDL, http://pshdl.org/), building and verifying new hardware (incl. debugging)
- Education, Courses

Instance 1 Instance 2 ... Complete (24 instances)



Resource usage on the FPGA chip (floorplan): with efficient HDL EAI template design + load balancing, UDFs as high-level synthesis become a dominating factor for multi-instance parallelism

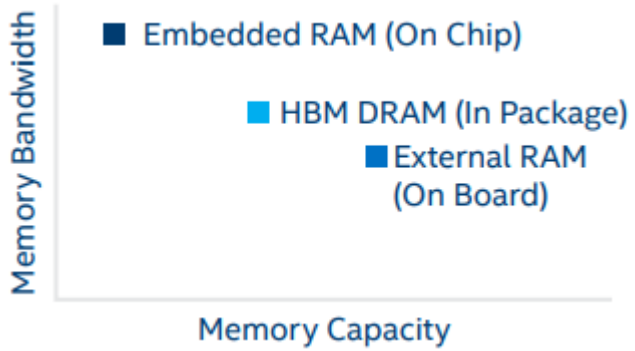
Programming Models

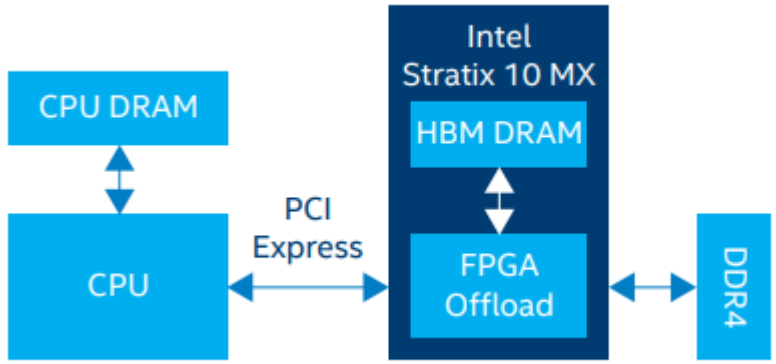
Memory Access / Bandwidth

- on-Chip memory accessible in few clock cycles
- Capacity of on-chip memory not enough (flip-flops often required for program logic)

Requires:

- Fast off-chip DRAM memory access (shared with CPU) from the FPGA
- (even Non-volatile Memory)
- Study of optimization teqhniques (e.g., message indexing [RRM2017] vs. streaming)





(e.g., Intel HBM2 https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/wp/wp-01264-stratix10mx-devices-solve-memory-bandwidth-challenge.pdf)

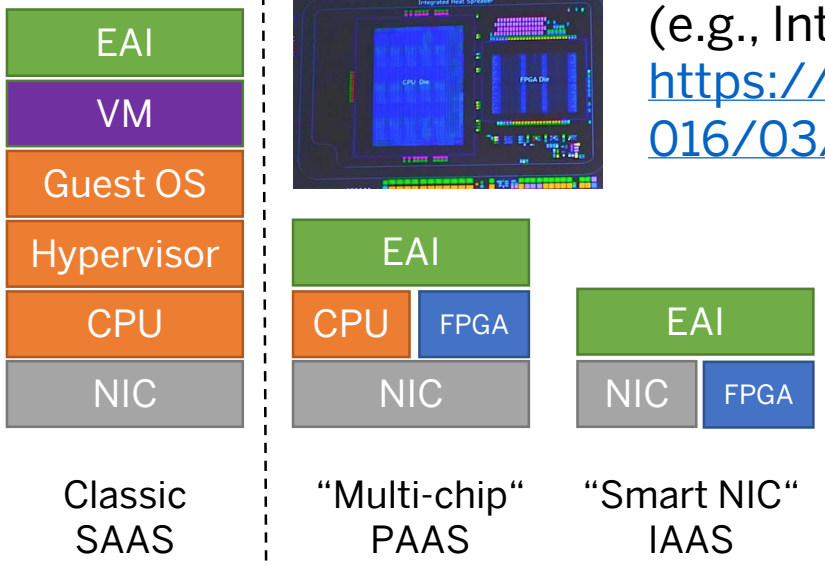
Programming / Architecture Models

Local Integration System, Network EAI

- ➕ closer to the network, combined data flow architecture and CPU
- ➖ Requires special HW: FPGA + x

Requires:
- FPGA+CPU multi-chip architectures with direct NIC access
- FPGA on NIC or SmartNIC

New EAI Architecture Variants



(e.g., Intel Broadwell, https://www.theregister.co.uk/2016/03/14/intel_xeon_fpga/)

Not just data pipes (logic pushdown)

Processing Models

Message Exchange Patterns

- ➕ FPGA works well for inOnly streaming
- ➖ In/out, request-reply in some integration scenarios

→ Requires:
- Streaming with request-reply (e.g., JMS asynchronous + correlation identifier)
- or Synch-Asynch Bridges (e.g., [RH2015])

Transport Protocol Support

- ➕ stateless transport protocols
- ➖ IP cores missing (even for UDP), no stateful transport protocols

→ Requires:
- Vendor IP core support for a broader coverage of protocols like TCP, HTTP, MQTT
- or efficient SW/HW co-design to leverage software protocol implementations (e.g., [YZXQFR2011]).

Non-functional aspects

- ➕ many built-in IP cores, e.g., for network, memory access
- ➖ vendor specific IP cores incomplete, e.g., for security

→ Requires:
- Vendor IP core support for non-functional aspects like different types of authentication, encryption
- or efficient SW/HW co-design to leverage software implementations

e.g., Solace‘s own network controller

(Cloud) Operations

Multi-tenancy

+ tenants separated on HW
− limited resources on partitioned chip, cross-tenant processing

→ Requires:
- FPGA HW virtualization à la “Configurable Cloud“[Cau2017]

Solace‘s cross virtual provider messaging

Microsoft MS Project Catapult

Data center impact

+ low energy, less space
− to be added to datacenter blueprints, good troubleshooting / debugging tools

→ Requires:
- Integration in current cloud platforms

amazon web services FPGA Developer AMI

HA/DR setup

+ less prone to failures
− HA requires tenant distributions across different HW, DR across HW and data centers with transactions, adhere to regulations (e.g., data protection)

→ Requires:
- Regulation-aware, abstracted
- configurable HA/DR capabilities

similar to Solace‘s HA/DR broker

In general: SDNs

Message Endpoints

<u>Applications</u>

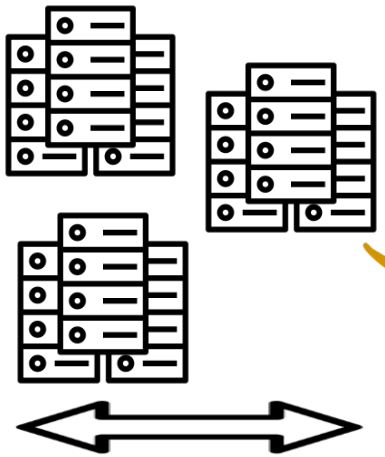
low latency, high-throughput

Endpoints limited capacities, discrepancy between EAI egress and endpoint ingress rates

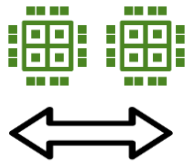
Requires:

- End-to-end flow control
- Application scaling
- Asynchronous message processing
- FPGA+RDMA
- ....

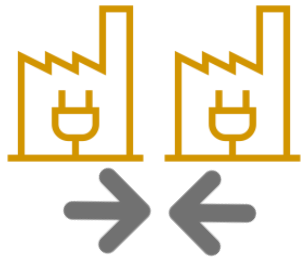
Disruption Potential



Run one integration scenario used by hundreds of customers on just few FPGAs (shared), run all scenarios of one customer on one FPGA, instead of hundreds of VMs

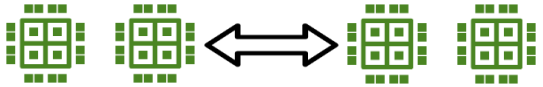


EAI System, Integration Processes



Reduce energy consumption, space

Stable, multi-tenant, HA+DR



New York (active)

Las Vegas (passive)

... more

New application programming models (e.g., asynch, more EAI logic -> idempotent, retries)

Conclusion

- Disruption through novel applications + EAI challenges and hardware + technology
- Specialization with reconfigurable hardware leads to promising future EAI architecture variants
- FPGAs are less well known and harder to program, while problem is not software engineers being able to program FPGAs, but eco-system required
- FPGAs allow for optimizations of both compute and I/O operations, data flow architecture -> think beyond the core application
- This is just the starting point: many new opportunities + further research

References (1/2)

[Cul1986] David E Culler. 1986. Dataflow architectures. Annual review of computer science 1,1 (1986),225–253.

[Lin2000] D. S. Linthicum. Enterprise Application Integration. Addison-Wesley, 2000.

[HW2004] Gregor Hohpe and BobbyWoolf.2004. Enterprise integration patterns: Designing, building, and deploying messaging solutions. Addison-Wesley.

[YZXQFR2011] Yu, J., Zhu, Y., Xia, L., Qiu, M., Fu, Y. and Rong, G., 2011, August. Grounding high efficiency cloud computing architecture: HW-SW co-design and implementation of a stand-alone Web server on FPGA. In *Applications of Digital Information and Web Technologies (ICADIWT), 2011 Fourth International Conference on the* (pp. 124-129). IEEE.

[LGMBEV2012] Lockwood, J.W., Gupte, A., Mehta, N., Blott, M., English, T. and Vissers, K., 2012, August. A low-latency library in FPGA hardware for high-frequency trading (HFT). In *High-Performance Interconnects (HOTI), 2012 IEEE 20th Annual Symposium on* (pp. 9-16). IEEE.

[RH2015] Ritter, D. and Holzleitner, M., 2015, June. Integration adapter modeling. In *International Conference on Advanced Information Systems Engineering* (pp. 468-482). Springer.

References (2/2)

[RMSRM2016] Daniel Ritter, Norman May, Kai Sachs, and Stefanie Rinderle-Ma. 2016. Benchmarking integration pattern implementations. In DEBS. 125–136.

[RMRM2017] Daniel Ritter, NormanMay, and Stefanie Rinderle-Ma. 2017. Patterns for emerging application integration scenarios: A survey. Information Systems 67(2017),36–57.

[Put2017] Andrew Putnam. The Configurable Cloud -- Accelerating Hyperscale Datacenter Services with FPGAs. Presentation at Active Workshop ICDE 2017.

[RDMRM2017] Daniel Ritter, Jonas Dann, Norman May, and Stefanie Rinderle-Ma. 2017. Hardware Accelerated Application Integration Processing: Industry Paper. In DEBS. 215–226.

[Cau2017] Adrian M. Caulfield, et al.: Configurable Clouds. IEEE Micro 37(3): 52-61 (2017).

[RRM2017] Daniel Ritter and Stefanie Rinderle-Ma: Toward Application Integration with Multimedia Data. In IEEE EDOC 2017.

Thank you

Contact information:

Daniel Ritter
daniel.ritter@sap.com

universität
wien