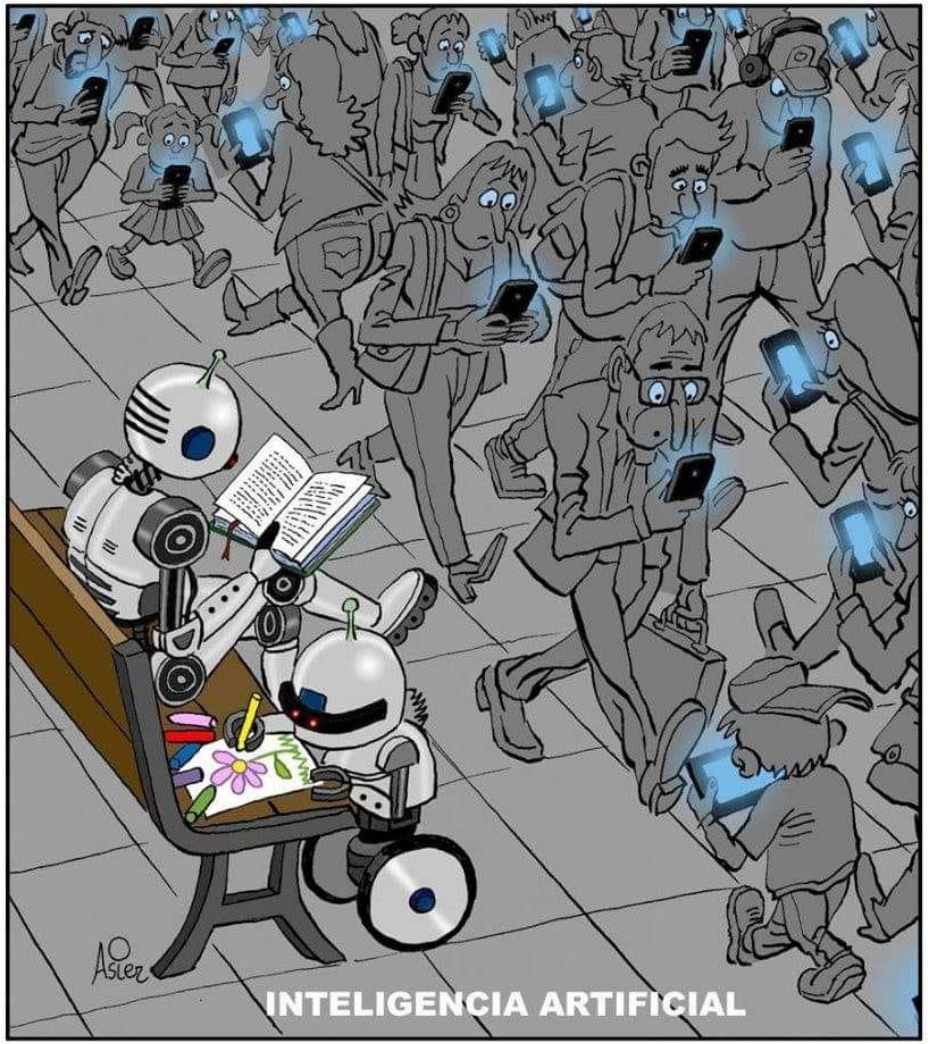
HPC, Big Data, and AI: Computing Under Constraints



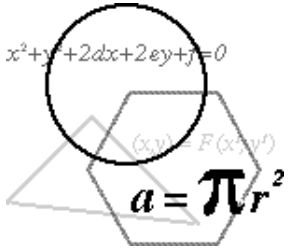
Dan Reed

dan.reed@utah.edu
www.hpcdan.org

@HPCDan

THE UNIVERSITY OF UTAH®

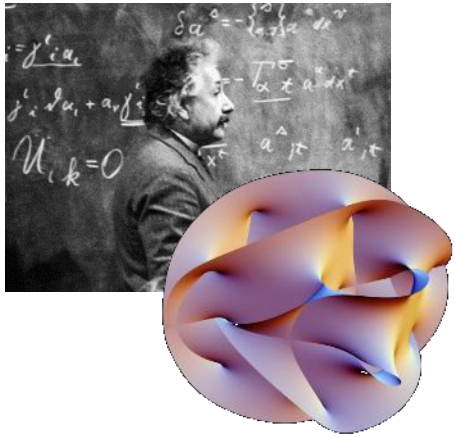
The changing nature of research and scholarship

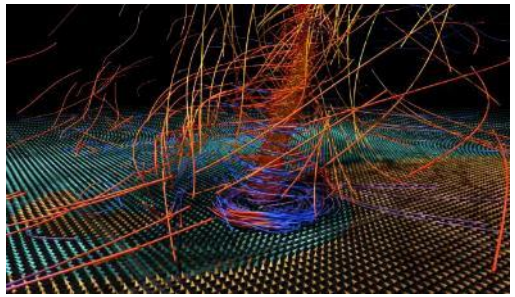


$$H(t)|\psi(t)\rangle = i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle$$

Experimental Theoretical Computational

Large Hadron Collider





Severe Storm Model (NCSA)

THE UNIVERSITY OF UTAH®

Science instrument continuum

BIG and FEW

Multidisciplinary Data Fusion & Modeling

SMALL and MANY

Vehicles

Array of Things

Health

THE UNIVERSITY OF UTAH®

Large Synoptic Survey Telescope (LSST)

Structure

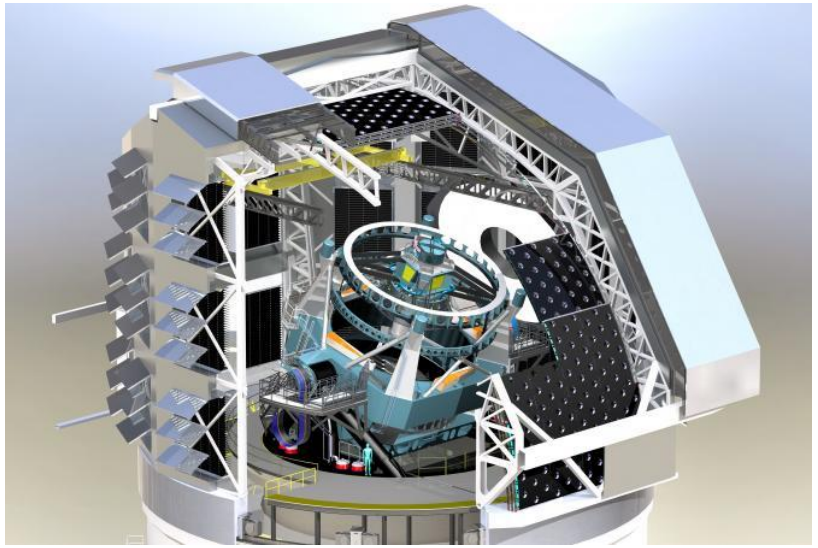
- 8.4 meter telescope with 3.5 degree FOV
- 3.2 gigapixel camera
- Construction underway in Chile

Science

- Dark matter/energy and gravitational lensing
- Outer solar system and NEOs
- Milky Way structure and evolution

Streamed foranalysis and access

- ***~15 TB/night and 200K images/year (1.3 PB)***
- ***60 second trigger for transient events***
- 60 PB raw and 15 PB catalog (over 10 years)



THE UNIVERSITY OF UTAH®

Everything gets smart at the edge …

Oura Ring

Oxford Nanopore
Mobile DNA sequencer

THE UNIVERSITY OF UTAH®

Changing the game

IBM's DeepBlue chess machine

- *Programmed for chess*

Google's AlphaGo

- *Generalized reinforcement learning*

Google's AlphaZero

- *Generative adversarial network*

AlphaStar and Starcraft II

MIT Technology Review

Artificial Intelligence

How AlphaZero has rewritten the rules of game play on its own

AI just won a poker tournament against professional players

sing cause
n Kaulen/Carnegie Mellon University

y Timothy Revell

n AI just claimed another gaming victory over humans by winning a 20-day poker tournament. The AI, called bratus, took on four of the world's best Heads-Up No-Limit Texas Hold 'Em poker players at a Pennsylvania casinc

Mastering the game of Go with deep neural networks and tree search

David Silver[1]*, Aja Huang[1]*, Chris J. Maddison[1], Arthur Guez[1], Laurent Sifre[1], George van den Driessche[1], Julian Schrittwieser[1], Ioannis Antonoglou[1], Veda Panneershelvam[1], Marc Lanctot[1], Sander Dieleman[1], Dominik Grewe[1], John Nham[2], Nal Kalchbrenner[1], Ilya Sutskever[2], Timothy Lillicrap[1], Madeleine Leach[1], Koray Kavukcuoglu[1], Thore Graepel[1] & Demis Hassabis[1]

Some panel framing questions

How do we best meet soft real-time constraints for streaming data, subject to energy, communication, operation and cost constraints?

What advances have we seen in orchestration capabilities for real-time considerations?

What AI techniques are relevant addressing real-time data constraints?

Can we use AI for performance and efficiency tuning of both HPC applications and the data center?

Some additional questions

What are the issues of greatest interest for big data management of HPC system monitoring data?

What applications are emerging that have the potential for strong societal impact?

How should HPC systems and data centers evolve to support AI applications?

Are AI specific hardware accelerators the only future?

Autonomous goal seeking entities, aka panel members

Rosa Badia
Barcelona Supercomputing Center

Charlie Catlett
Argonne National Laboratory

Ewa Deelman
USC/Information Sciences Institute

Greg Koenig
KPMG

Satoshi Matsuoka
RIKEN Center for Computational Science

THE UNIVERSITY OF UTAH®

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

PyCOMPSs approach towards computing under constraints

Rosa M. Badia

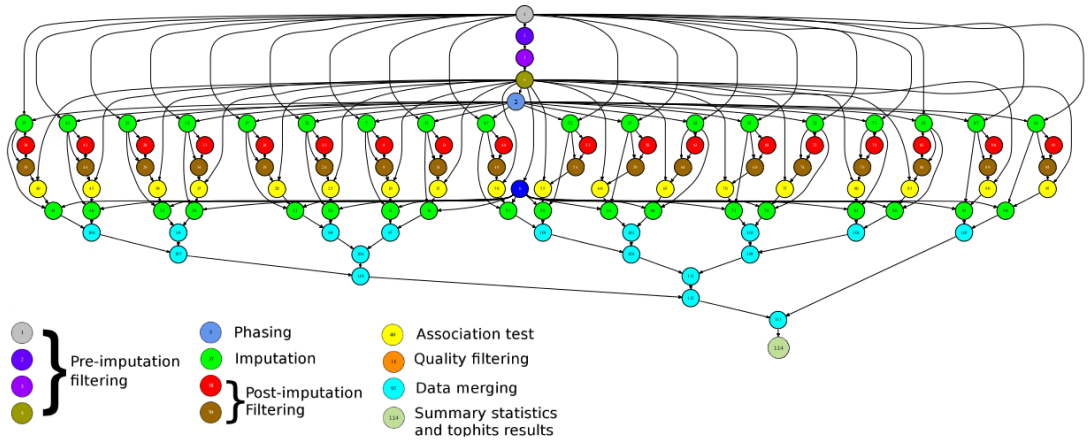
Workflows & Distributed Computing Group

22/11/2019 Denver (CO) USA

HPC Big Data and AI: Computing under Constraints

Programming with PyCOMPSs/COMPSs

- Sequential programming, parallel execution
- General purpose programming language + annotations/hints
  - To identify tasks and directionality of data
- Builds a task graph at runtime that express potential concurrency
- Offers a shared memory illusion to applications in a distributed system
  - The application can address larger data storage space: support for Big Data apps
  - Support for persistent storage
- Agnostic of computing platform
  - Enabled by the runtime for clusters, clouds and container managed clusters



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

What is a PyCOMPSs/COMPSs task?

- Tasks can be sequential, multi-threaded, multi-node
- Tasks may have constraints/requirements
- Tasks may have multiple versions
- **Tasks may read/write streamed data**

```
@implement (source class="myclass", method="myfunc")
@constraint (MemorySize=1.0, ProcessorType ="ARM")
@task (c=INOUT)
def myfunc_other (a, b, c):
    ...
```

```
@constraint (computingUnits= "248")
@mpi (runner="mpirun", computingNodes= "16", ...)
@task (returns=int, stdOutFile=FILE_OUT_STDOUT, ...)
def nems(stdOutFile, stdErrFile):
    pass
```

Hybrid Task Flows and Data Flows: new types of pipelines

```
@task(fds=STREAM_OUT)
def sensor(fds):
    while not end():
        data = get_data_from_sensor()
        f.write(data)
        f.flush()
    fds.close()
```

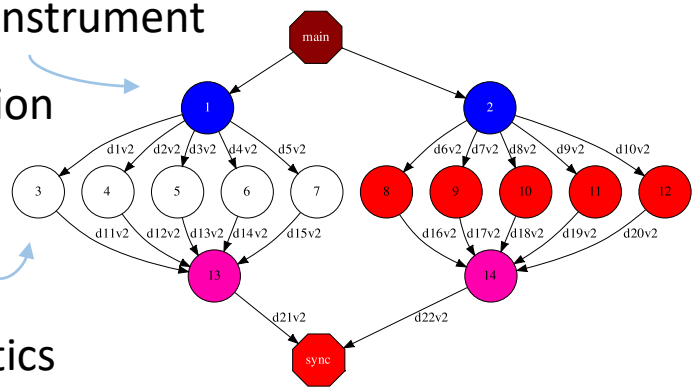
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Use case: Continuous data generation and analytics

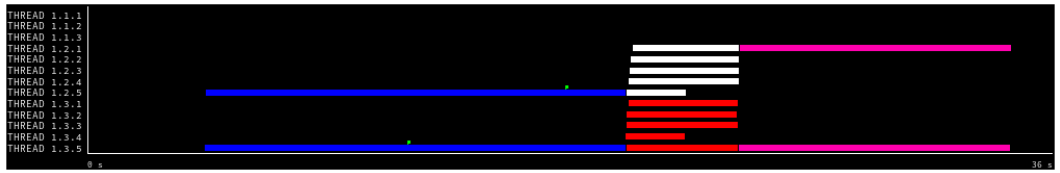
- Sample case: Sensor/instrument generating streamed data to be processed by some analytics

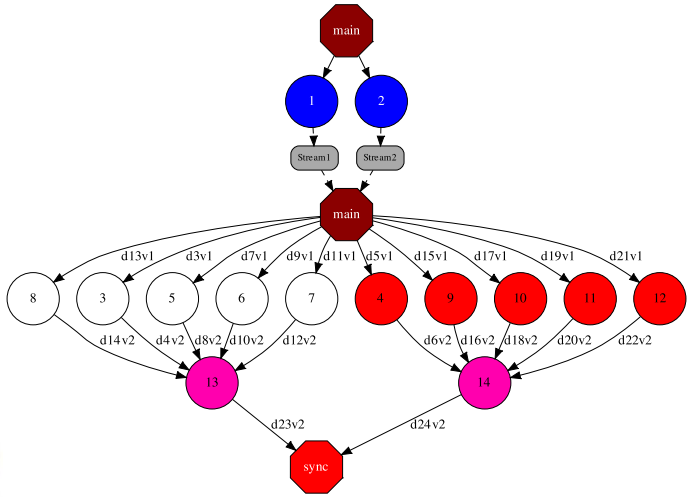
Sensor/instrument

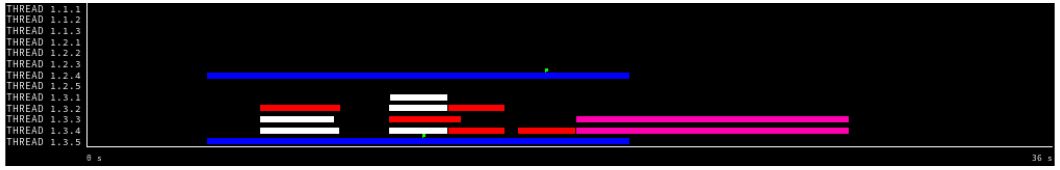
…or MPI simulation



Analytics







Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Task groups and timeouts

- Timeouts can be defined for each tasks
- Tasks can raise exceptions

```
@task(file_path=FILE_IN, time_out=200)
def time_out_task (file_path):
    ...
```

```
@task(file_path=FILE_INOUT)
def comp_task(file_path):
    ...
    raise COMPSsException("Exception raised")
```

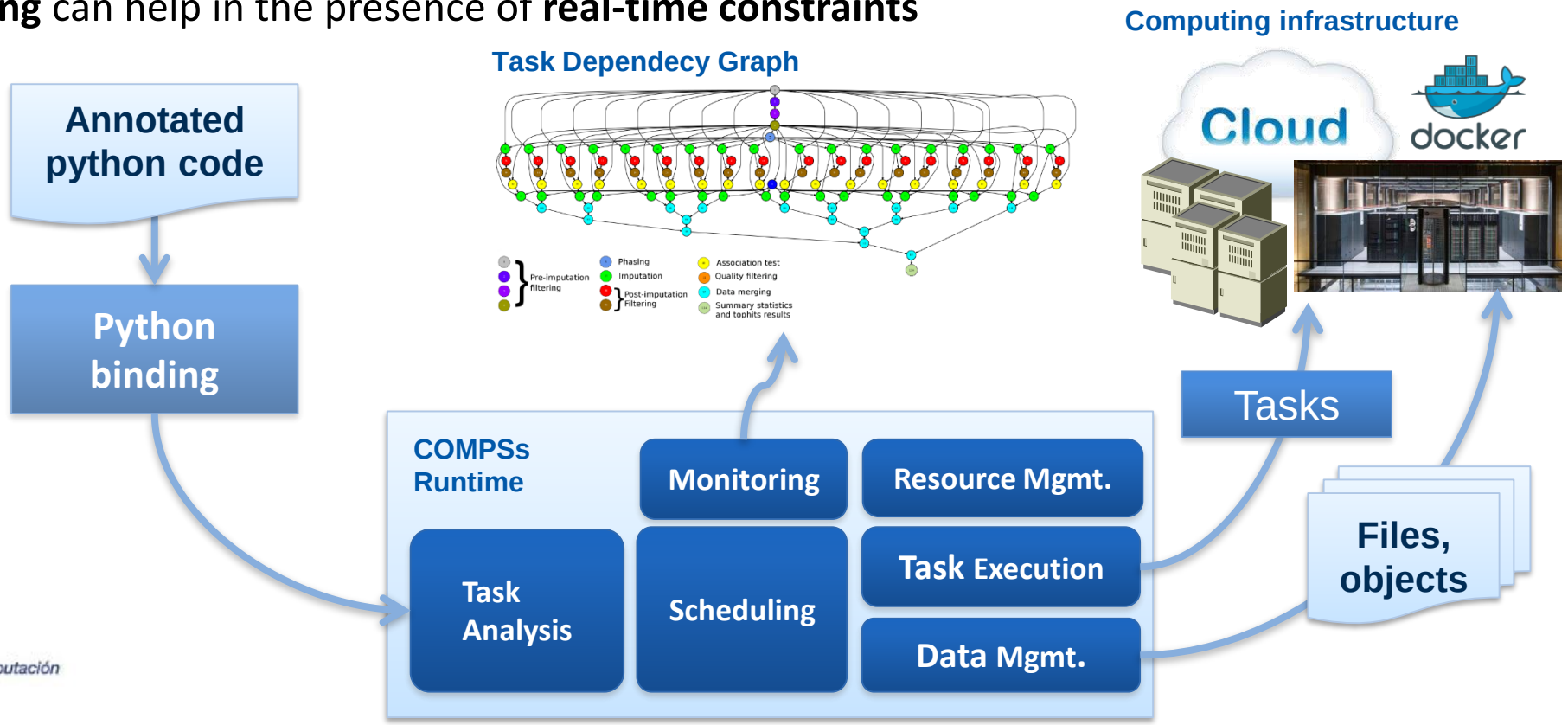
- Combined with groups of tasks enables to cancel the group of tasks on the occurrence of an exception
- Can be combined with streamed data to dynamically make decisions depending on the actual behavior

```
def test_cancellation(file_name):
    try:
        with TaskGroup('failedGroup'):
            long_task(file_name)
            long_task(file_name)
            executed_task(file_name)
            comp_task(file_name)
    except COMPSsException:
        print("COMPSsException caught")
        write_two(file_name)
    write_two(file_name)
```

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

COMPSs runtime: Static scheduling / dynamic scheduling

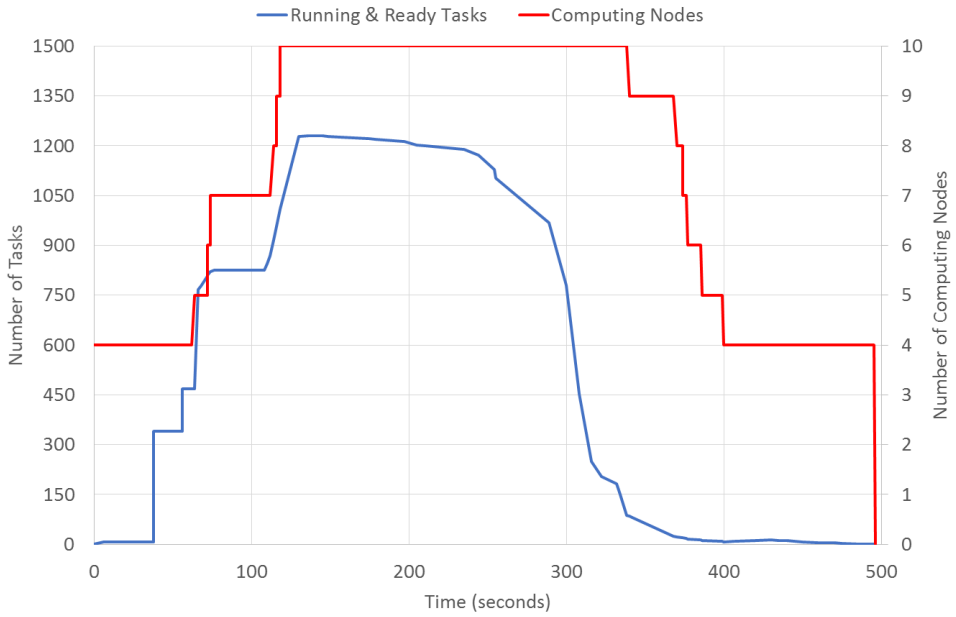
- PyCOMPSs/COMPSs applications executed in distributed mode following the master-worker paradigm
  - Description of computational infrastructure in an XML file
- Sequential execution starts in master node and tasks are offloaded to worker nodes
- All data scheduling decisions and data transfers are performed by the runtime
- **Static scheduling** can help in the presence of **real-time constraints**

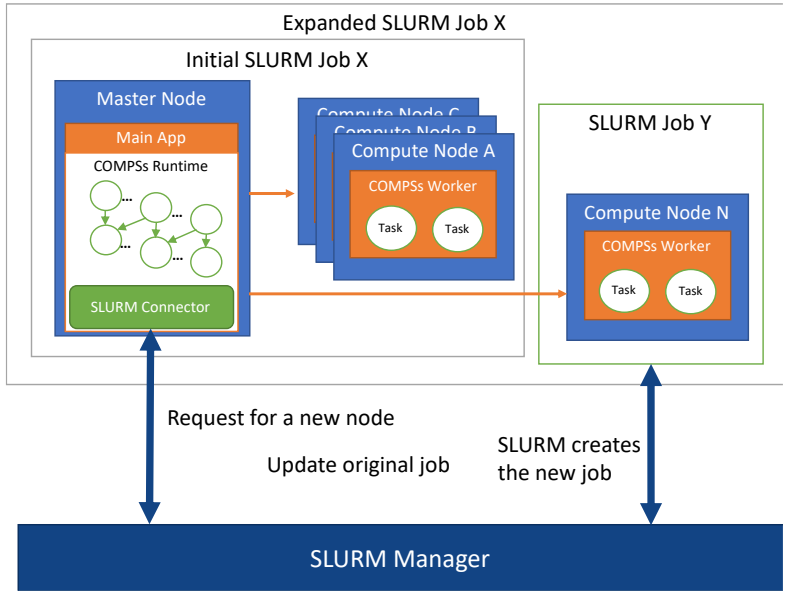


Barcelona Supercomputing Center
Centro Nacional de Supercomputación

COMPSs runtime: support for elasticity

- Possibility to adapt the computing infrastructure depending on the actual workload
- Now also for cloud, also SLURM managed systems
- Feature that contributes to a more effective use of resources
- Current elasticity policy is based on cost or time
  - **Can be extended to be based on real-time constraints**

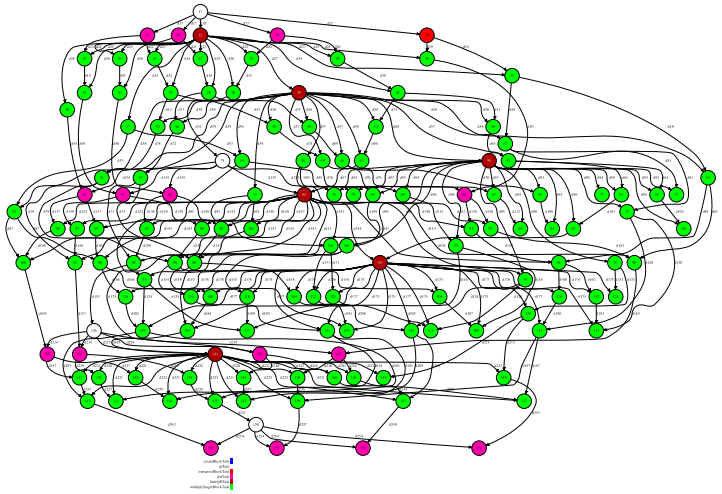




Barcelona Supercomputing Center
Centro Nacional de Supercomputación

TANGO

PyCOMPSs/COMPSs ambition



- Complex infrastructures and applications
- Holistic approach where both data and computing are integrated in a single flow built on simple, high-level interfaces
  - Integration of computational workloads, with machine learning and data analytics
  - Intelligent runtime that can make scheduling and allocation, data-transfer, and other decisions
- **New applications with social impact**
  - Digital twins
  - Urgent computing
  - Personalized medicine and medical software platforms
  - Precision agriculture

Cloud

HPC
Exascale computing

Sensors,
Instruments
Edge devices

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

Thank you

support-compss@bsc.es

rosa.m.badia@bsc.es

Big data behind the big data behind the big data

"…Extensive instrumentation, monitoring and control of ~~high-performance computing~~ systems and their data-centers is the big data behind big data…."

~~distributed~~ urban and environmental

Charlie Catlett
Senior Computer Scientist
University of Chicago and
Argonne National Laboratory

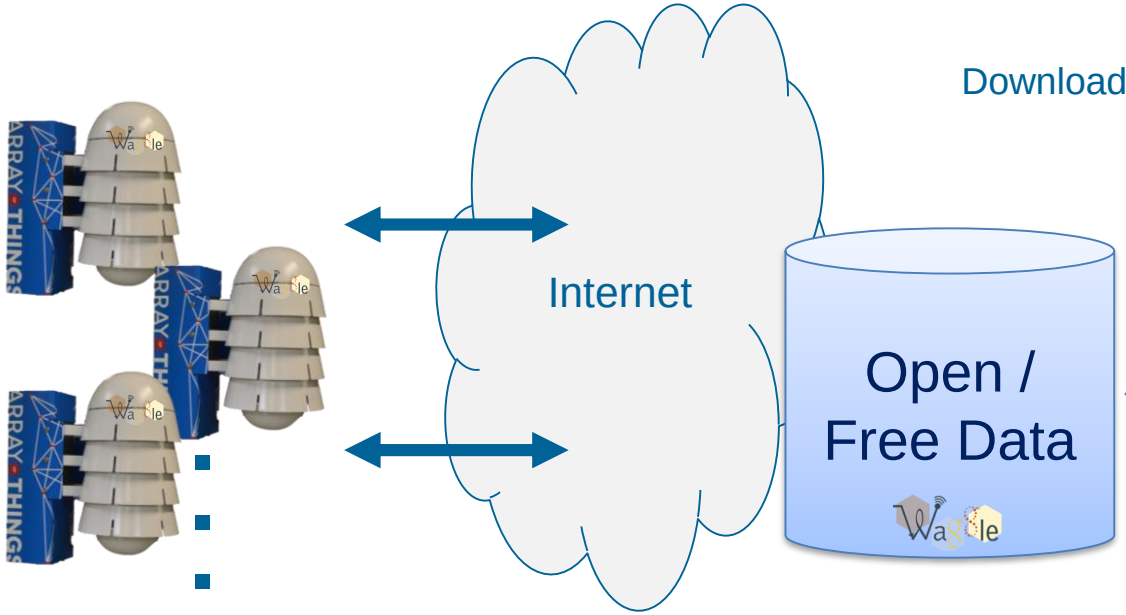
November 2019

THE UNIVERSITY OF CHICAGO

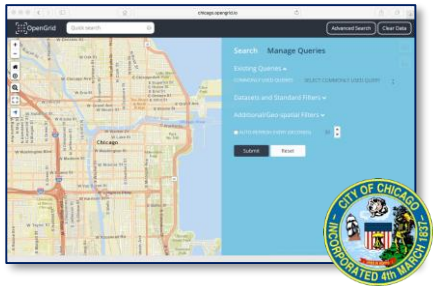
Argonne NATIONAL LABORATORY

catlett@anl.gov

Making data available… but what about remote sources?



Download: Scientific Analysis



Portals: Data Discovery

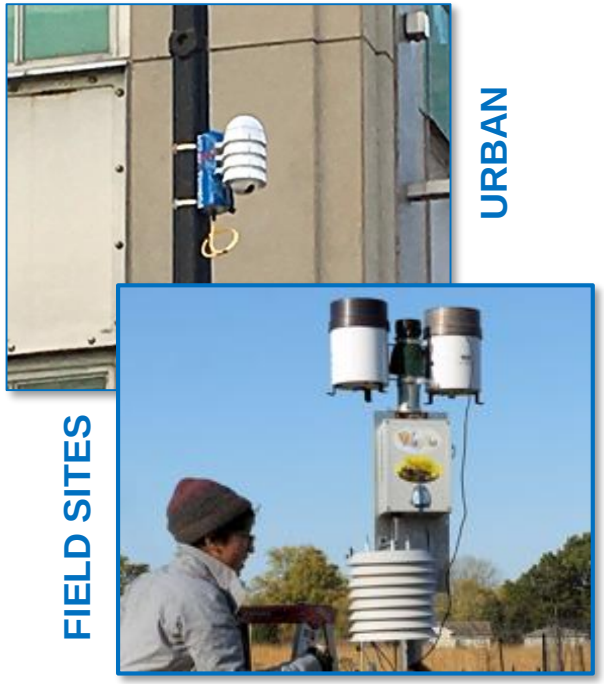
Tools and Tutorials

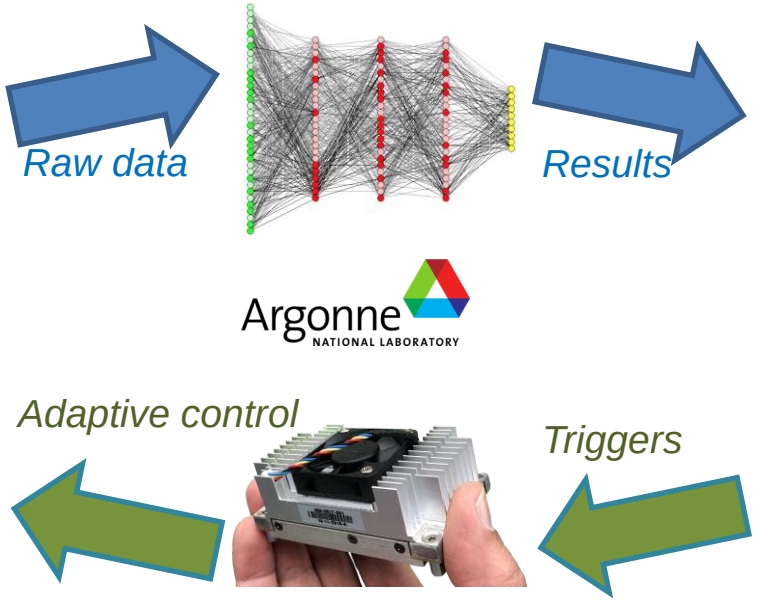
Real-Time Access: New Applications

Why not stream everything all the time?

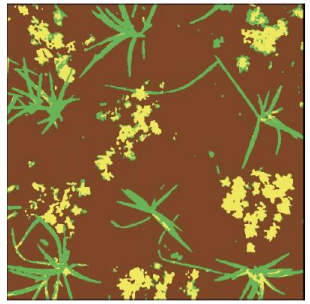
- Insufficient (or expensive) bandwidth
- Privacy and ethics
- *Software-defined sensors*
- Real-time adaptive sensing and/or actuation

Software-Defined Sensors

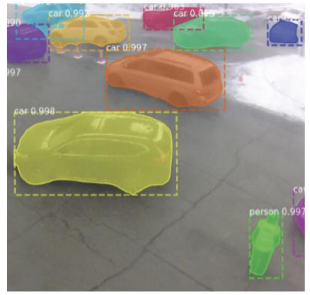




Plant Species



Pedestrian Flow



Traffic Flow

Wildlife

Autonomous Adaptive Sensing

Pete Beckman, Charlie Catlett (ANL)

Northwestern University
THE UNIVERSITY OF CHICAGO
Argonne NATIONAL LABORATORY

Software-Defined Sensors: Traffic and Pedestrian Flows

Nicola Ferrier, Songha Park, Yeongho Kim (ANL)

Northwestern University
THE UNIVERSITY OF CHICAGO
Argonne NATIONAL LABORATORY

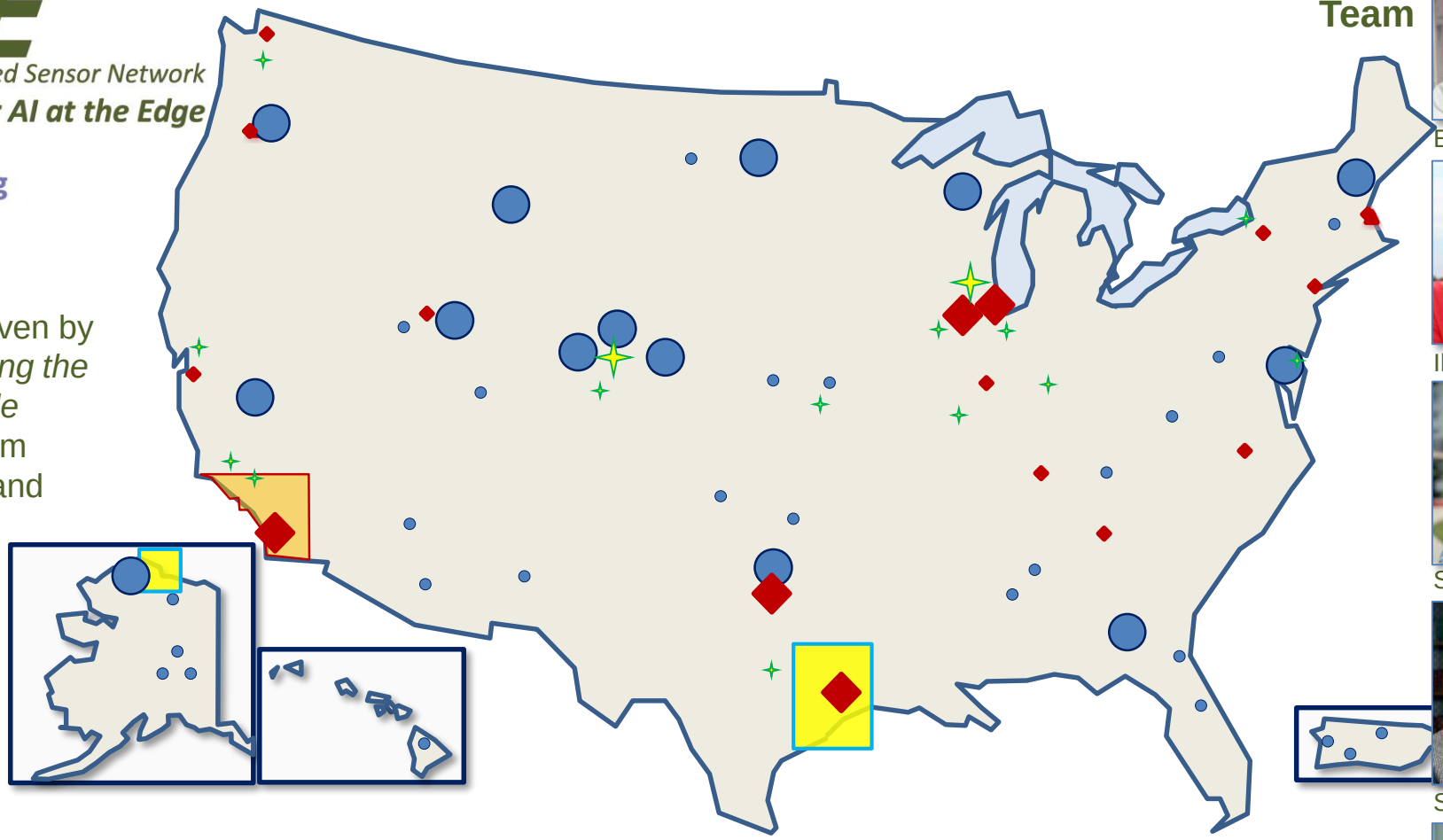
SAGE

A Software-Defined Sensor Network

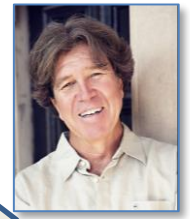
Cyberinfrastructure for AI at the Edge

www.sagecontinuum.org

Research cyberinfrastructure driven by questions related to *understanding the resilience of our nation at multiple spatial and temporal scales*—from extreme weather affecting food and water supplies, to gentrification impacting neighborhood cohesion, to urbanization disrupting ecological diversity.



Leadership Team


Eugene Kelly

Pete Beckman

Ilkay Altuntas

Charlie Catlett

Scott Collis

Valerie Taylor

Stacie Williams

Dan Reed

Jim Olds

Irene Qualters

Triggered Computation, Deep Learning Training

HPC / Cloud

NEON **WIFIRE/ HPWRE** **ARM** **AoT** **UWI**

SAGE is funded by the National Science Foundation's Mid-Scale Research Infrastructure program. SAGE is led by Northwestern University with partners at the University of Chicago, Argonne National Laboratory, Colorado State University, University of California-San Diego, Northern Illinois University, George Mason University, University of Utah, DePaul University, Lincoln Park Zoo Urban Wildlife Initiative, and over 20 collaborating universities around the world.

USC Viterbi School of Engineering
Information Sciences Institute

Ewa Deelman, Ph.D.

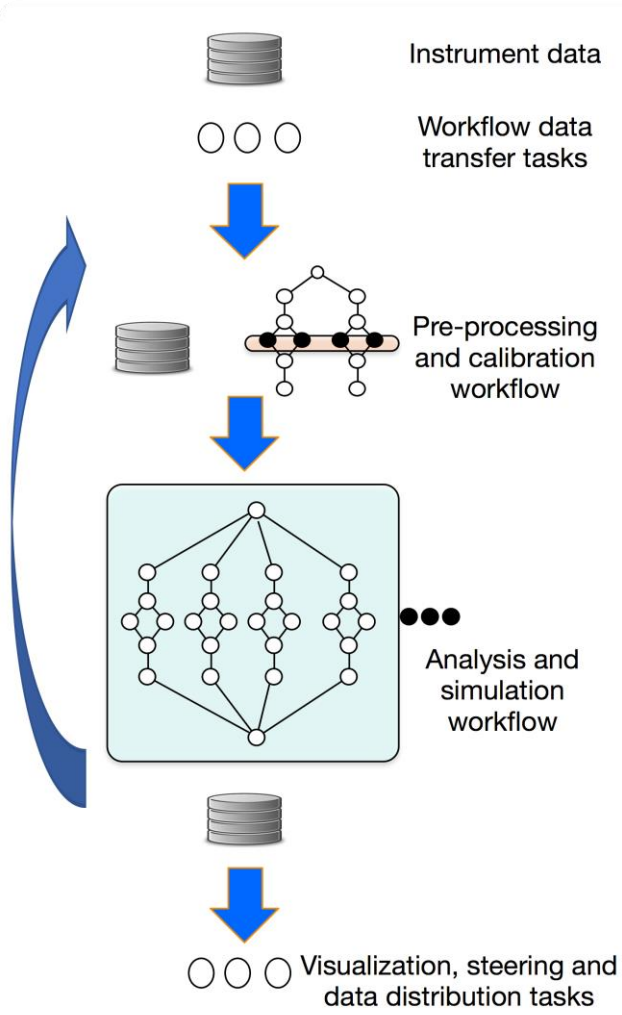
University of Southern California,

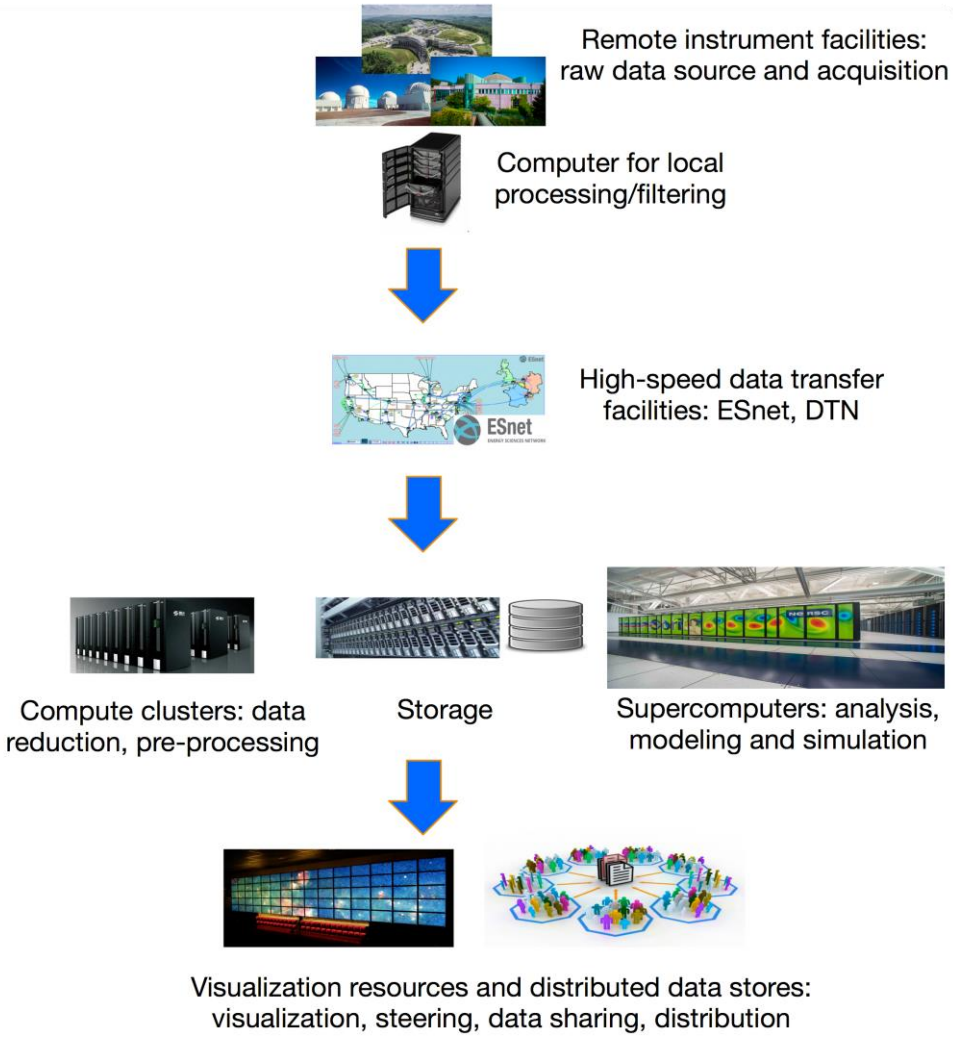
Information Sciences Institute

Can we use AI for performance and efficiency tuning on both HPC applications and the data-center?

SC'19 HPC Big Data and AI: Computing under Constraints
Denver, CO, November 20, 2019

Scientific Workflow Operate Across CI

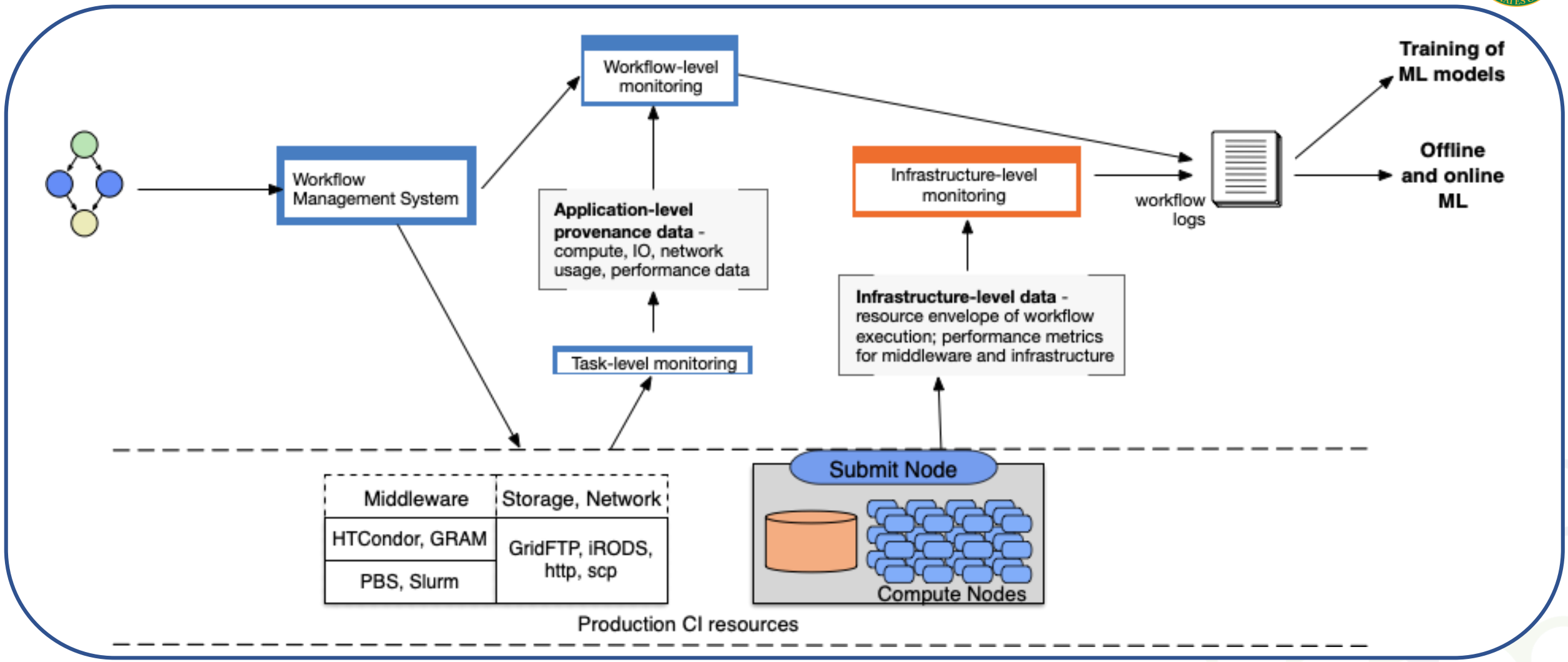




Automating Complex Workflow Execution

- Resource discovery
- Mapping tasks to resources
- Manage data staging in and out
- Automate execution of complex task graphs
- Automate error/anomaly detection and recovery
- Automate performance and reliability optimizations

Workflow Executions Generate Big Data



AI for Science Process Automation

CI generation

- Fully autonomous workflow that builds up the CI to operate an instrument and conducts the experiment

Data Generation

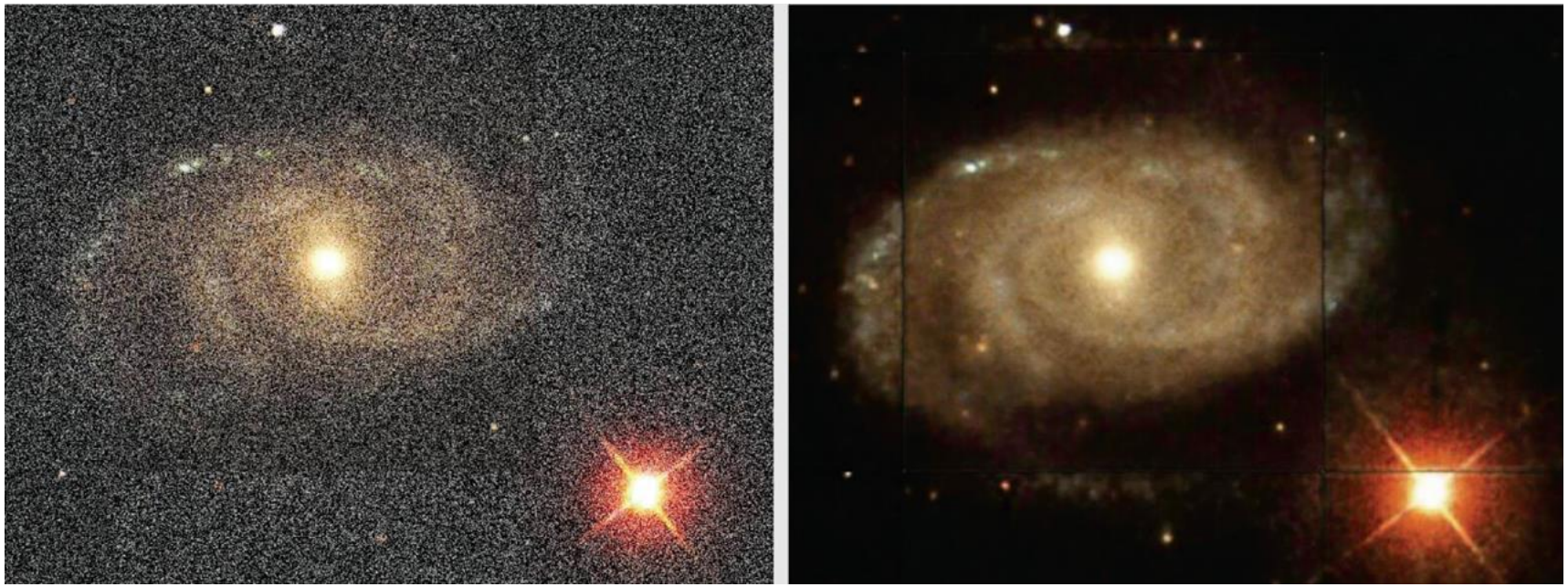
- Fully autonomous workflow that discovers the needed instruments, services, data, composes them, conducts the experiment

Science generation

- Fully autonomous workflow that builds an instrument, discovers the needed services, data, composes them, conducts the experiment

How to discern truth?

AI that "knows" what a galaxy should look like transforms a fuzzy image (left) into a crisp one (right).



Generative adversarial networks

Generator: makes up images

Discriminator tries to identify manipulations

Generator gets better

Thousands of real images of galaxies artificially degraded

PHOTOS (LEFT TO RIGHT): SDSS AND CE ZHANG (ETHZ) (CC-BY); KEVIN SCHAWINSKI/ETH ZURICH/INSTITUTE FOR ASTRONOMY

Increased use of automation and ML presents a new set of challenges

Delivering usable systems is hard, decreases usability, increases risk, decreases reliability

- Trust: How do you know that what we observe is real?
- Transparency
- Understanding
- Reproducibility

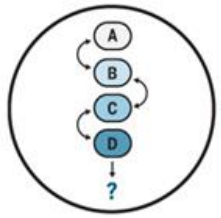
Can AI take over scientific research?

1. Explore the scientific literature
Find the most relevant papers in a sea of millions, track new topics as they emerge.

Semantic Scholar
A search engine that extracts not just words from papers, but graphs and "influential" citations.

Iris.AI
A browsing tool for exploring scientific papers by the concepts that link them.

2. Design experiments
Find the right trade-off between exploration of new ground and exploitation of well-trodden phenomena.



Zymergen
A company with an AI that tracks thousands of variables while tweaking microbe genomes (see main story, p. 18).

3. Run experiment
Keep track of thousands of tiny tubes, molecules, and cells, minimizing the imprecision and mistakes that ruin careers.

Transcriptic, Emerald Cloud Lab
Cloud-based robotic laboratories for remotely doing automated molecular and cellular biology experiments.

4. Interpret data
Make sense of the flood of genetic and biochemical results that now flow from biological experiments.

Nutonian
A software platform that ingests very large data sets and spits out a mathematical theory that explains the patterns in the data.

5. Write scientific paper
So far the closest thing to a paper-writing AI is a postdoc. But even writing papers can be enhanced with software that can read the draft of your paper.

Citeomatic
A free online tool that reads your paper and predicts what citations are missing.

NEWS

The cyberscientist

John Bohannon
+ See all authors and affiliations

Science 07 Jul 2017:
Vol. 357, Issue 6346, pp. 18-21
DOI: 10.1126/science.357.6346.18

- So what will be the role of the human?
- How do we re-invent ourselves in the era of automation?
- How will being a scientist differ in the future?
- How do we create new human/machine interactions (intentions/expectations/level of trust)?

HPC Big Data and AI: Computing Under Constraints

SC19 Panel Discussion

Denver, Colorado – November 22, 2019

Gregory A. Koenig (koenig@acm.org)

Resources that might be constrained

Computational Resources

- Number of CPU cores
- Memory
- Communication bandwidth

Data Resources

- Limitations on the amount and/or quality of input data
- Limitations on data motion

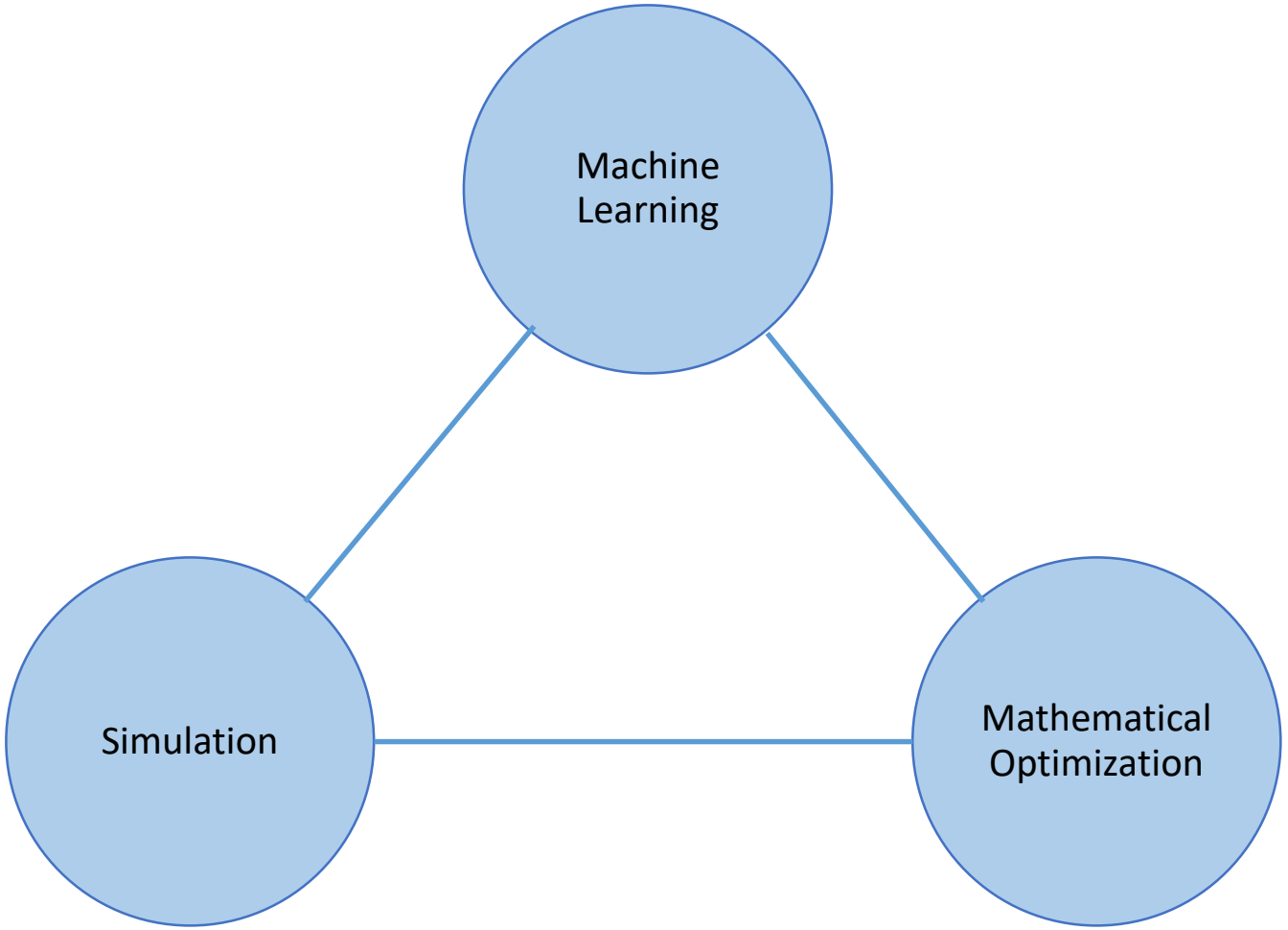
Time Resources

- Deadline to reach a solution
- Communication latency

Power/Energy Resources

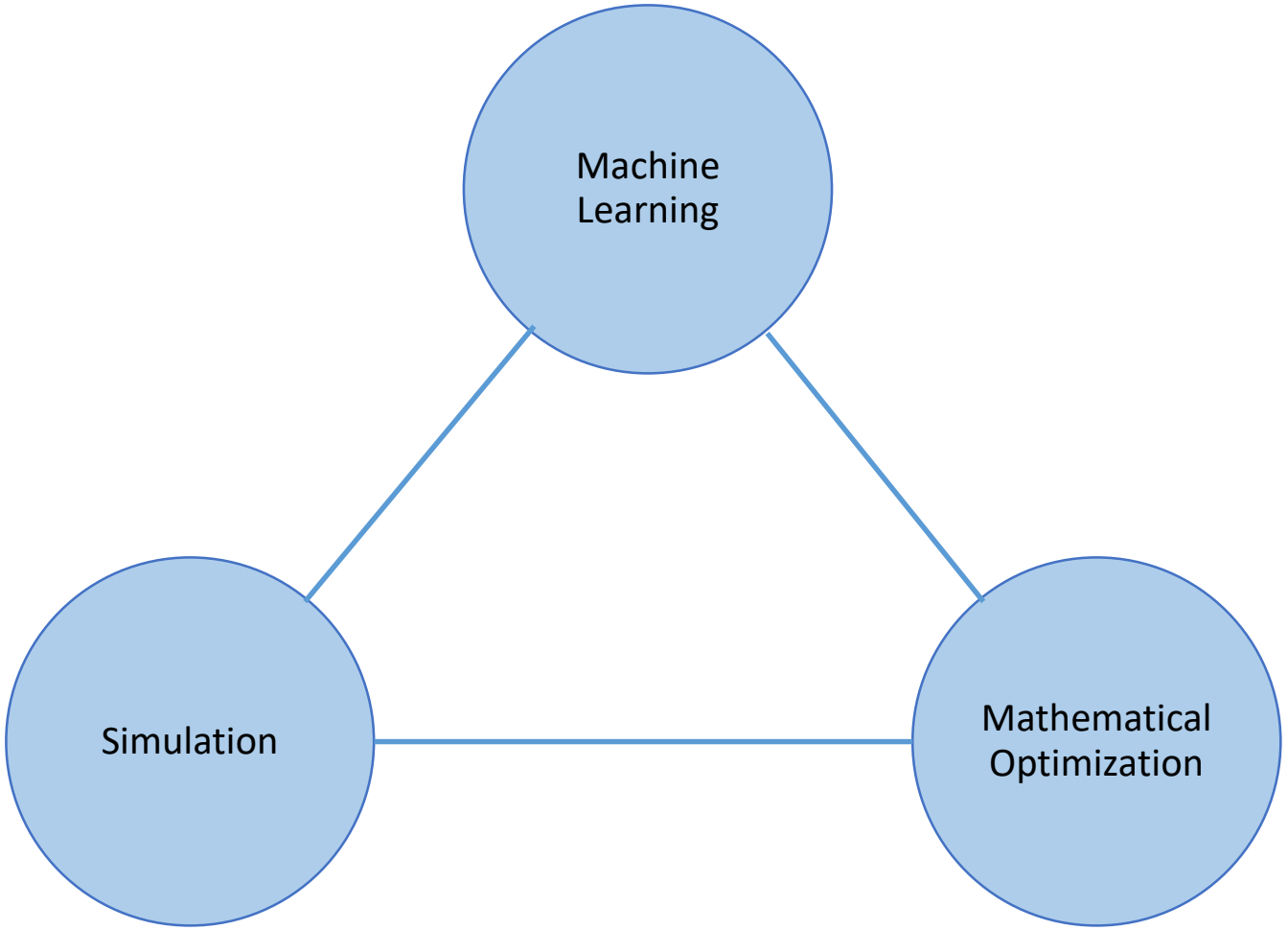
- Limitations on the maximum total instantaneous power draw
- Constraints on the total energy available during some period of time (e.g., one day)

AI techniques



AI techniques

- Statistical Methods
  - Data constraints (training)
- Deep Networks
  - Data constraints (training)
  - Computational constraints

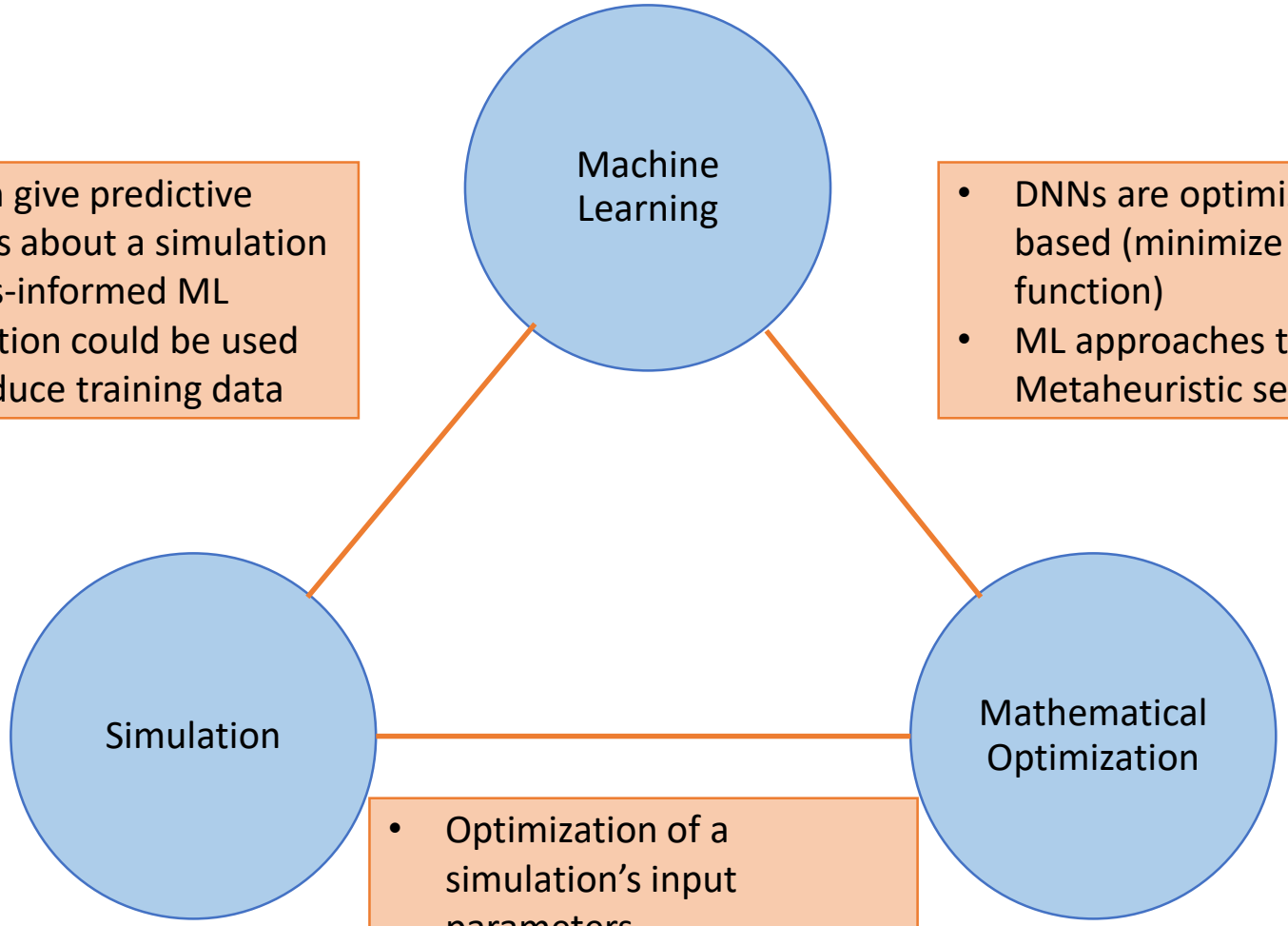


- Scientific Simulation
  - Computation constraints
  - Time constraints
  - Data constraints
- Monte Carlo and others
  - Computational constraints

- Exact Solvers
  - Time constraints
- Metaheuristics
  - Time constraints
  - Computational constraints

AI techniques

- Statistical Methods
  - Data constraints (training)
- Deep Networks
  - Data constraints (training)
  - Computational constraints



- ML can give predictive insights about a simulation
- Physics-informed ML
- Simulation could be used to produce training data

- DNNs are optimization based (minimize loss function)
- ML approaches to guide Metaheuristic searches

- Scientific Simulation
  - Computation constraints
  - Time constraints
  - Data constraints
- Monte Carlo and others
  - Computational constraints

- Optimization of a simulation's input parameters
- Optimal real-time computation steering

- Exact Solvers
  - Time constraints
- Metaheuristics
  - Time constraints
  - Computational constraints

AI techniques

- Statistical Methods
  - Data constraints (training)
- Deep Networks
  - Data constraints (training)
  - Computational constraints

<u>Key take-away:</u> It is often possible to use techniques from one AI domain to address constraints in a connected domain.

- Scientific
  - Computation constraints
  - Time constraints
  - Data constraints
- Monte Carlo and others
  - Computational constraints

Optimization

- Optimization of a simulation's input parameters
- Optimal real-time computation steering

- Time constraints
- Metaheuristics
  - Time constraints
  - Computational constraints

Thank you!

Gregory A. Koenig (koenig@acm.org)

The first “exascale” supercomputer Fugaku – HPC, BD & AI

- *Satoshi Matsuoka*
- *Director, RIKEN Center for Computational Science*

- **20191121 SC19 Panel @ Denver**

The ‘Fugaku’ Supercomputer, Successor to the K-Computer
Installation Dec. 2019~, operations early 2021

Arm64fx & Fugaku 富岳 /Post-K are:

- **Fujitsu-Riken design A64fx ARM v8.2 (SVE), 48/52 core CPU**
  - *HPC Optimized:* Extremely high package high memory BW (1TByte/s), on-die Tofu-D network BW (~400Gbps), high SVE FLOPS (~3Teraflops), various AI support (FP16, INT8, etc.)
  - Gen purpose CPU – Linux, Windows (Word), other SCs/Clouds
  - Extremely power efficient – > **10x power/perf efficiency for CFD benchmark** over current mainstream x86 CPU
- **Largest and fastest supercomputer to be ever built circa 2020**
  - > 150,000 nodes, superseding LLNL Sequoia
  - > 150 PetaByte/s memory BW
  - Tofu-D 6D Torus NW, 60 Petabps injection BW (10x global IDC traffic)
  - 25~30PB NVMe L1 storage
  - The first 'exascale' machine (not exa64bitflops =>apps perf.)

Green500, Nov. 2019

A64FX prototype -

Fujitsu A64FX 48C 2GHz

ranked #1 on the list

768x general purpose A64FX CPU w/o accelerators

- 1.9995 PFLOPS @ HPL, 84.75%
- 16.876 GF/W
- Power quality level 2

The GREEN 500

HOME GREEN500 LISTS RESOU

Home / Lists / November 2019

NOVEMBER 2019

Green500

GREEN500 LIST

EXCEL FILE

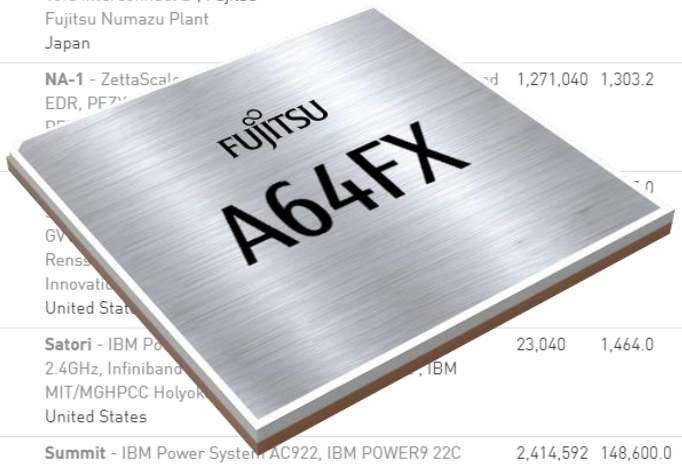
- The most energy-efficient system and No. 1 on the Green500 is a new Fujitsu A64FX prototype installed at Fujitsu, Japan. It achieved 16.9 GFlops/Watt power-efficiency during its 2.0 Pflop/s Linpack performance run. It is listed on position 160 in the TOP500.
- In second position is the NA-1 system, a PEZY Computing / Exascaler Inc. system which is currently being readied at PEZY Computing, Japan for a future installation at NA Simulation in Japan. It achieve 16.3 GFlops/Watt power efficiency. It is on position 421 in the TOP500.
- The No 3 on the Green500 is AiMOS, a new IBM Power systems at the Rensselaer Polytechnic Institute Center for Computational Innovations (CCI), New York, USA. It achieved 15.8 GFlops/Watt and is listed at position 25 in the TOP500.

Green500 List for November 2019

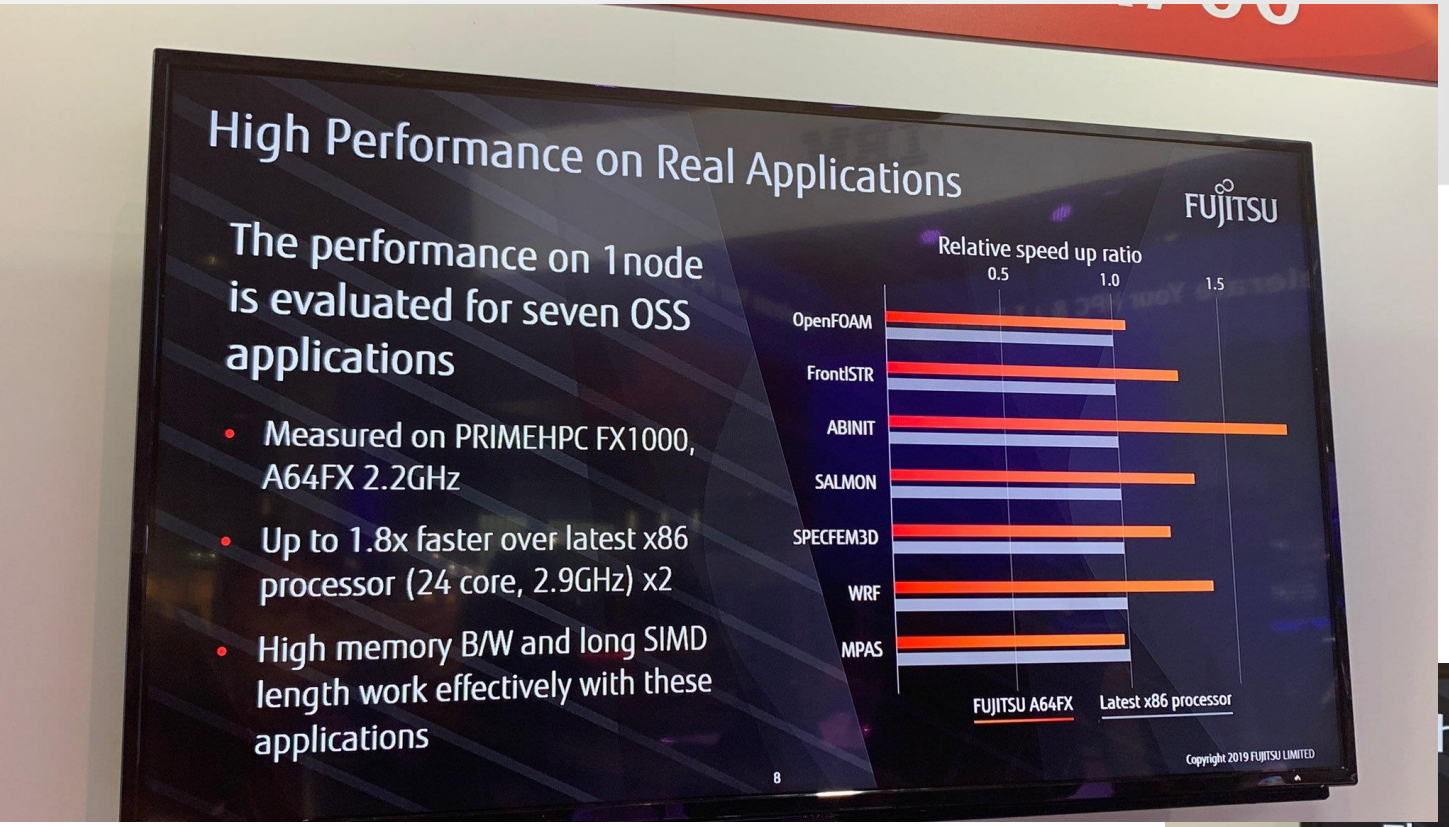
Listed below are the November 2019 The Green500's energy-efficient supercomputers ranked from 1 to 10.

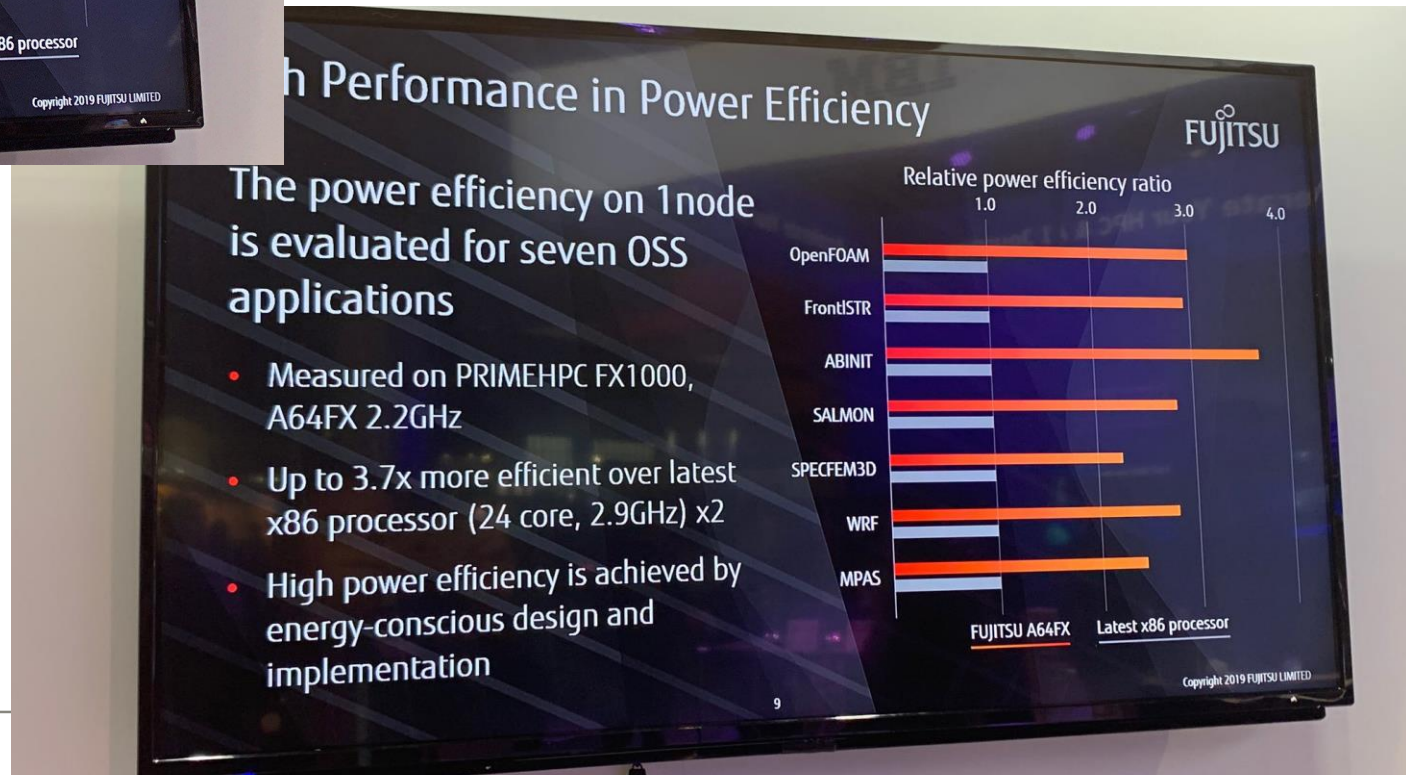
Note: Shaded entries in the table below mean the power data is derived and not meassured.

Rank	TOP500 Rank	System	Cores	Rmax (TFlop/s)	Power (kW)	Power Efficiency (GFlops/watts)
1	159	**A64FX prototype** - Fujitsu A64FX, Fujitsu A64FX 48C 2GHz, Tofu interconnect D , Fujitsu Fujitsu Numazu Plant Japan	36,864	1,999.5	118	16.876
2	420	**NA-1** - ZettaScal EDR, PE	1,271,040	1,303.2	80	16.256
3	24	GV Renss Innovati United Stat		0	510	15.771
4	373	**Satori** - IBM Po 2.4GHz, Infiniband MIT/MGHPCC Holyok United States	23,040	1,464.0	94	15.574
5	1	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR	2,414,592	148,600.0	10,096	14.719



Copyright 2019 FUJITSU LIMITED


High Performance on Real Applications
FUJITSU
The performance on 1node is evaluated for seven OSS applications
Measured on PRIMEHPC FX1000, A64FX 2.2GHz
Up to 1.8x faster over latest x86 processor (24 core, 2.9GHz) x2
High memory B/W and long SIMD length work effectively with these applications
Relative speed up ratio
0.5
1.0
1.5
OpenFOAM
FrontISTR
ABINIT
SALMON
SPECFEM3D
WRF
MPAS
FUJITSU A64FX
Latest x86 processor
Copyright 2019 FUJITSU LIMITED
8


h Performance in Power Efficiency
FUJITSU
The power efficiency on 1node is evaluated for seven OSS applications
Measured on PRIMEHPC FX1000, A64FX 2.2GHz
Up to 3.7x more efficient over latest x86 processor (24 core, 2.9GHz) x2
High power efficiency is achieved by energy-conscious design and implementation
Relative power efficiency ratio
1.0
2.0
3.0
4.0
OpenFOAM
FrontISTR
ABINIT
SALMON
SPECFEM3D
WRF
MPAS
FUJITSU A64FX
Latest x86 processor
Copyright 2019 FUJITSU LIMITED
9

Pursuing Convergence of HPC & AI (1)

- **Acceleration of Simulation (first principles methods) with AI (empirical method) :** ***AI for HPC***
  - Interpolation & Extrapolation of long trajectory MD
  - Reducing parameter space on Paretho optimization of results
  - Adjusting convergence parameters for iterative methods etc.
  - AI *replacing* simulation when exact physical models are unclear, or excessively costly to compute
- **Acceleration of AI with HPC:** ***HPC for AI***
  - HPC Processing of training data -data cleansing
  - Acceleration of (Parallel) Training: Deeper networks, bigger training sets, complicated networks, high dimensional data…
  - Acceleration of Inference: above + real time streaming data
  - Various modern training algorithms: Reinforcement learning, GAN, Dilated Convolution, etc.

R-CCS Pursuit of Convergence of HPC & AI (2)

- **Acceleration of Simulation (first principles methods) with AI (empirical method) : *AI for HPC***
  - ***Most*** R-CCS research & operations teams investigating use of AI for HPC
  - 9 priority co-design issues area teams also extensive plans
  - Essential to deploy AI/DL frameworks efficiently & at scale on A64fx/Fugaku

- **Acceleration of AI with HPC: *HPC for AI***
  - New teams instituted in Science of Computing to accelerate AI
    - Kento Sato (High Performance Big Data Systems)
    - Satoshi Matsuoka (High Performance AI Systems)
    - Masaaki Kondo Next Gen (High Performance Architecture)
  - **NEW: Optimized AI/DL Library via port of DNNL (MKL-DNN)**
    - **Arm Research + Fujitsu Labs + Riken R-CCS + others**
    - **First public ver. by Mar 2020, TensorFlow, PyTorch, Chainer, etc.**

Massive Scale Deep Learning on Fugaku

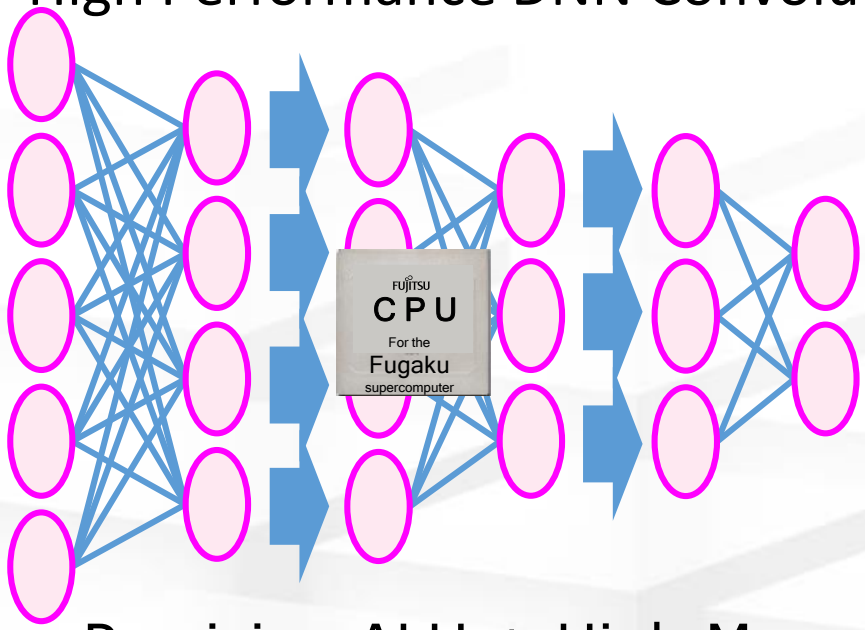
Fugaku Processor

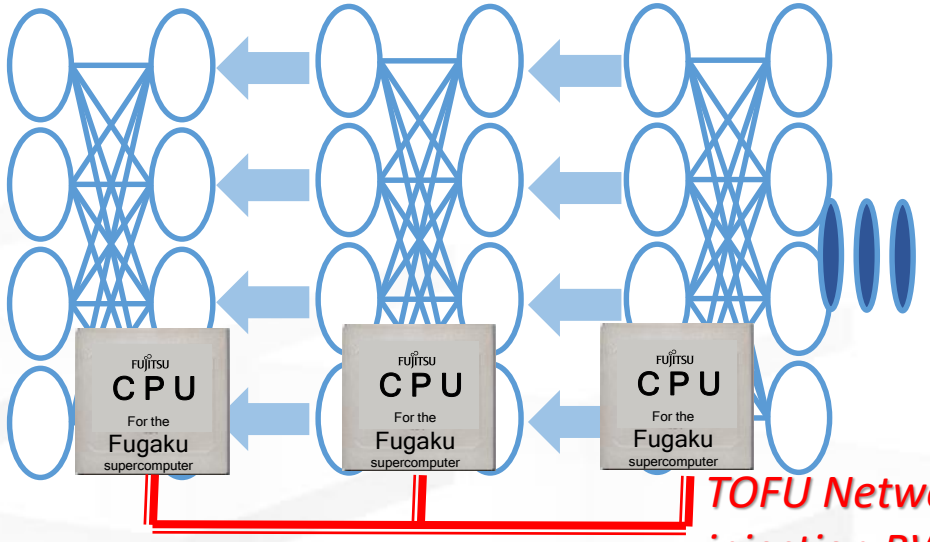
- ◆High perf FP16&Int8
- ◆**High mem BW for convolution**
- ◆**Built-in scalable Tofu network**

Unprecedened DL scalability

High Performance and Ultra-Scalable Network for massive scaling model & data parallelism

High Performance DNN Convolution





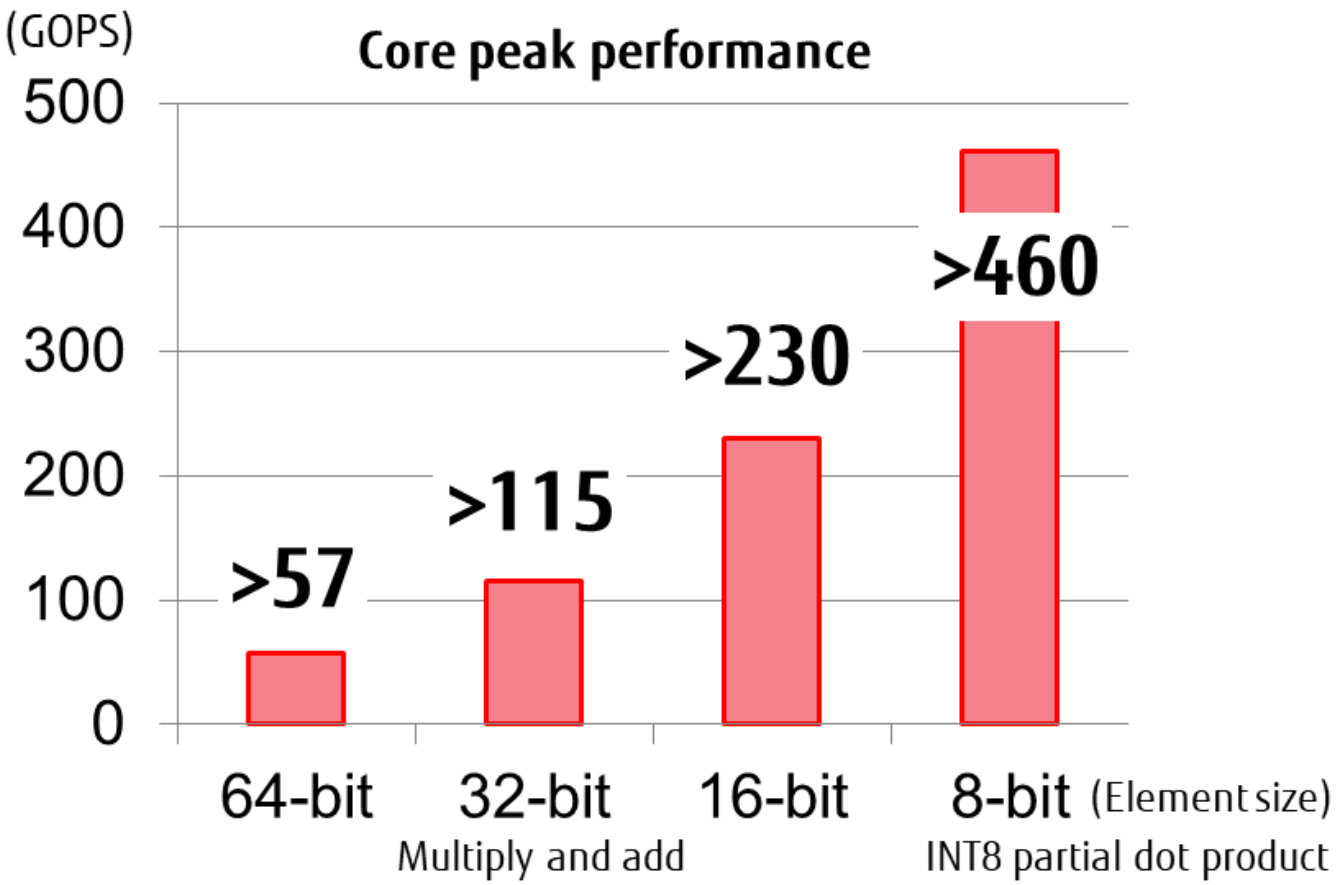
TOFU Network w/high injection BW for fast reduction

Low Precision ALU + High Memory Bandwidth + Advanced Combining of Convolution Algorithms (FFT+Winograd+GEMM)

Unprecedented Scalability of Data/

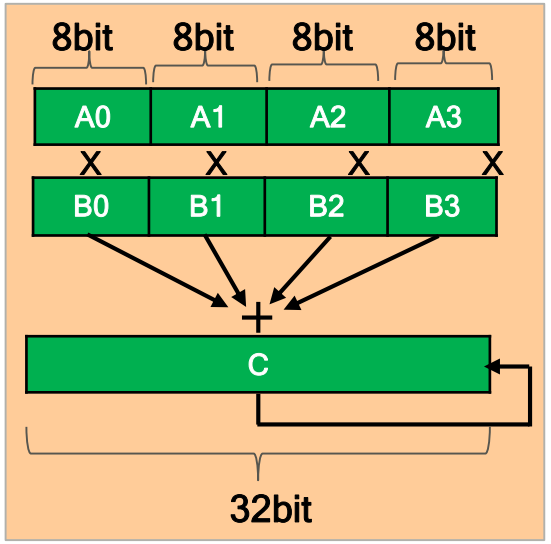
A64FX technologies: Core performance

- High calc. throughput of Fujitsu's original CPU core w/ SVE
  - 512-bit wide SIMD x 2 pipelines and new integer functions



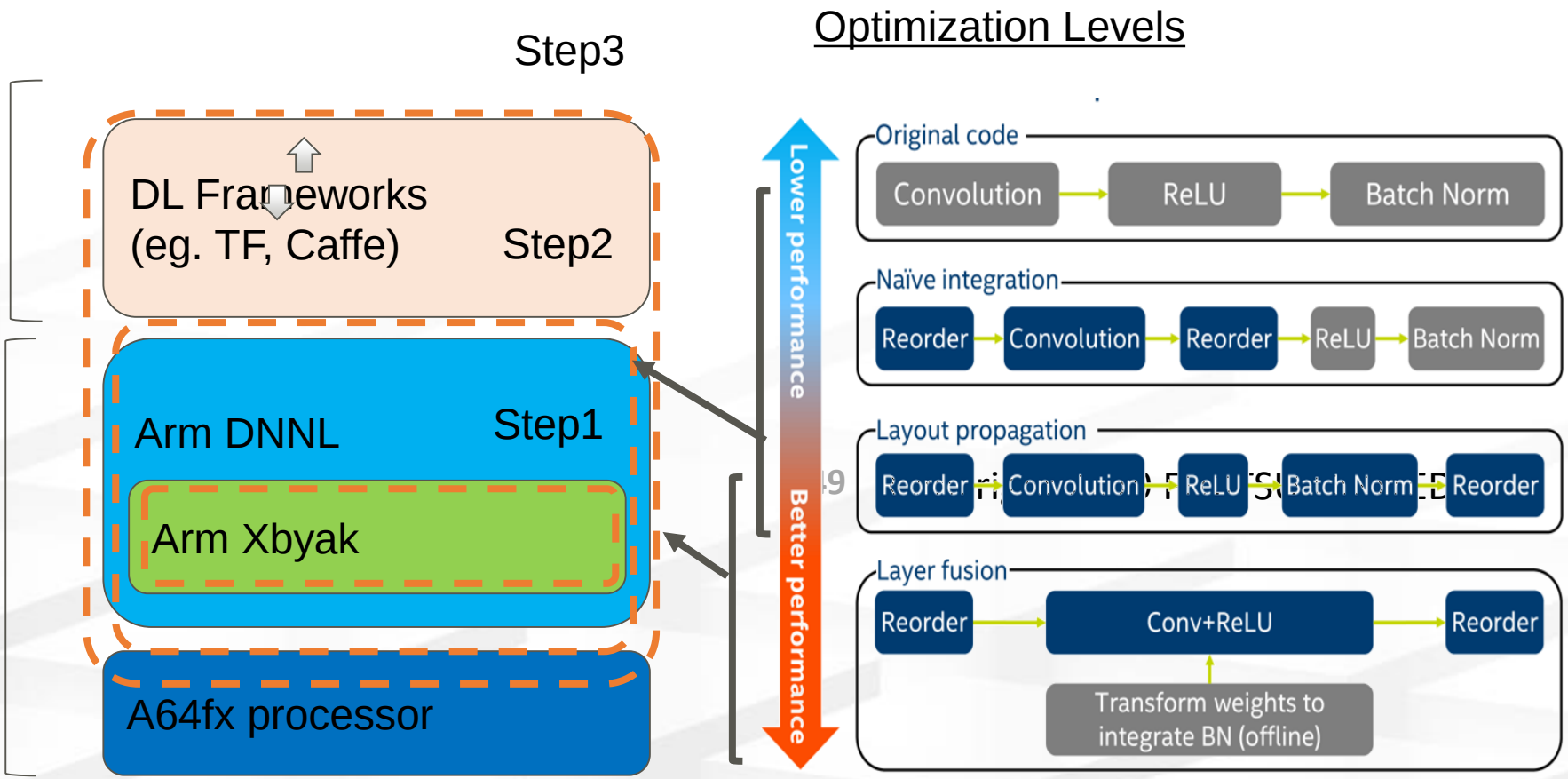
INT8 partial dot product

$C = \Sigma\ (Ai \times Bi) + C$



© 2019 FUJITSU

Fujitsu-Riken-Arm-Linaro joint effort on AI framework development on SVE/A64FX



Exaops of sim, data, and AI on Fugaku and Cloud

Fundamental Transformation of Big Data From 'Storing' to 'Processing'

- **Exponential Growth of Data worsen with IoT**
  - Current: Cloud => Edge, IoT Future Edge => Cloud
- **There is no way to *store* most of the data, we have to throw them away**
  - Filtering, in flight from edge => Cloud
  - Compression, in flight and analysis, application specific
  - Synthesis through simulation and AI Surrogates
- **Claim: Most big data observed just need statistical characterization; where actual 'data' is necessary, they can be generated from tiny data**
  - Days of 'MapReduce' will be gone RSN

Will HPC/AI Architectures Converge or Diverge

- **Converge**
  - High Bandwidth in Memory and Interconnect
  - Fast storage for temporary store
  - HPC for AI, AI for HPC in SW stack, applications
- **Diverge**
  - Precision
    - HPC: moderate to high (FP32-FP64), AI: moderate to low (FP32-INT4)
  - Arithmetic intensity
    - HPC: low to moderate, AI: moderate (large CNN filters) to high (transformers)