Perspectives on Neuromorphic Computing

Todd Hylton

Brain Corporation

hylton@braincorporation.com

ORNL Neuromorphic Computing Workshop

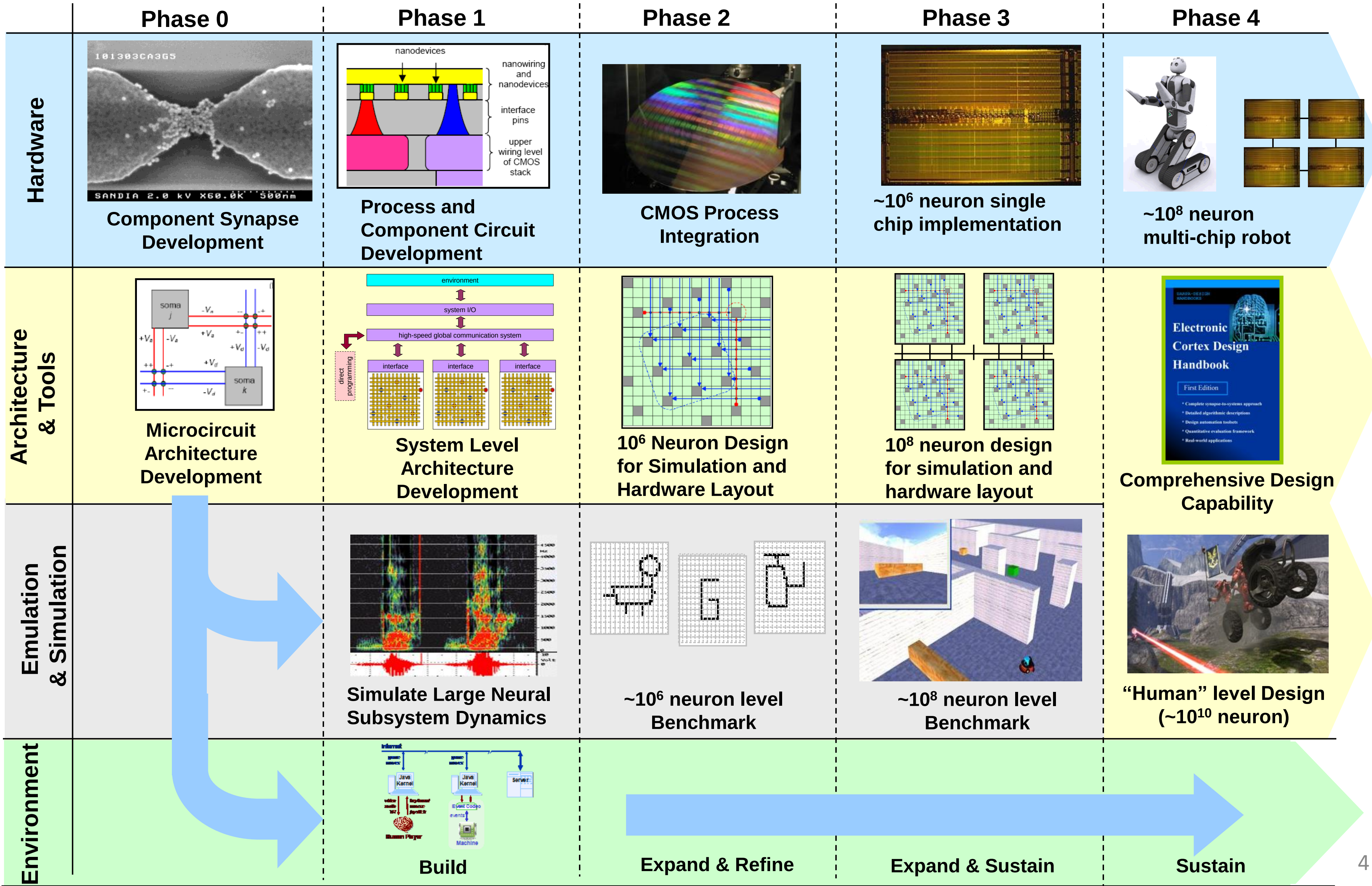
June 29, 2016

Outline

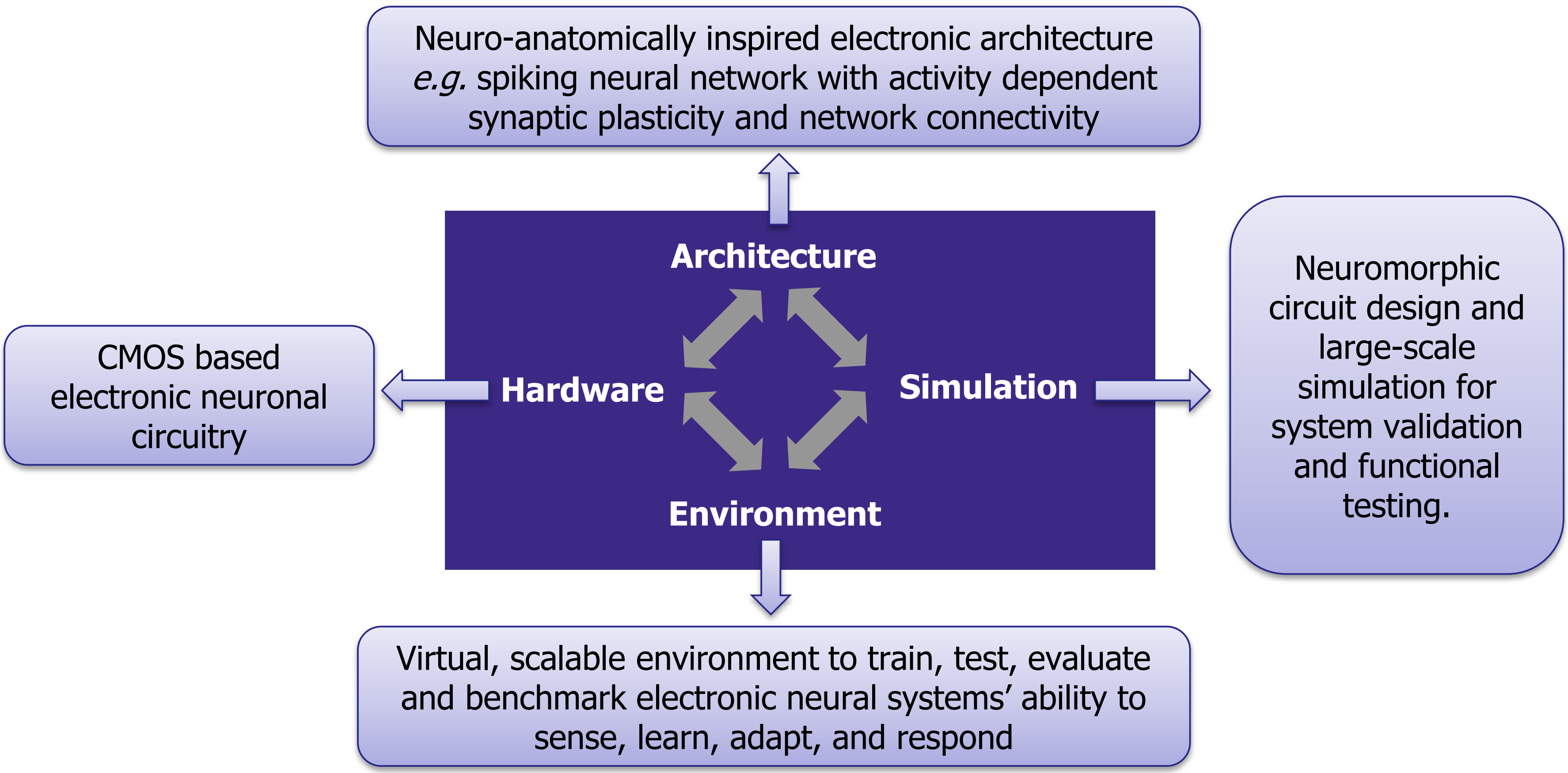
- Retrospective
  - SyNAPSE
- Perspective
  - Neuromorphic computing today
- Prospective
  - Technology landscape
  - Framing the opportunity
  - Goals for the future

Retrospective - SyNAPSE

SyNAPSE Program Plan



SyNAPSE Program Approach



Key Technology Issues / Choices

- Distributing large amounts of memory (synapses) among many processors (neurons) on a single chip.
  - Off-chip memory burns power and taxes memory bandwidth
  - DRAM needs large array sizes to be space efficient and does not integrate into most logic processes
  - Back end memory technologies (e.g. memristors, PCM) are immature and not available in SOA CMOS
- Developing a scalable messaging (spiking) architecture.
- Selection of computational primitives (e.g. neuron and synapse models)
- Engineering for scale, space and power efficiency
- Creating a large-scale simulation capability that accurately models the neuromorphic hardware
- Creating tools to develop and debug neural algorithms on the simulator and the neuromorphic hardware
- Writing the algorithms (including those that learn)

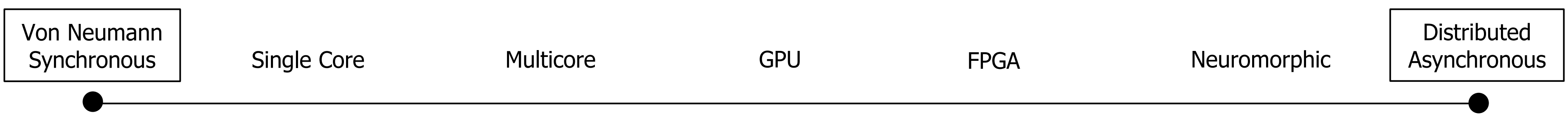
SyNAPSE – Miscellaneous Lessons Learned

- There are many, many ways to build a neuromorphic computer
- Although much can be leveraged from conventional computing technologies, building a neuromorphic computer requires a large investment in development tools
- Neuromorphic chip function can be replicated on a conventional computer, but with much lower efficiency.
- Biological scale networks are not only possible, but inevitable.
- The technology issues are challenging but surmountable.
- The time scale for developing a new memory technology and integrating it into SOA CMOS process is much longer than that needed to build a neuromorphic computer.
- The biggest current challenge in neuromorphic computing is creating the algorithms.

Perspective - Neuromorphic Computing Today

What is a Neuromorphic Computer?

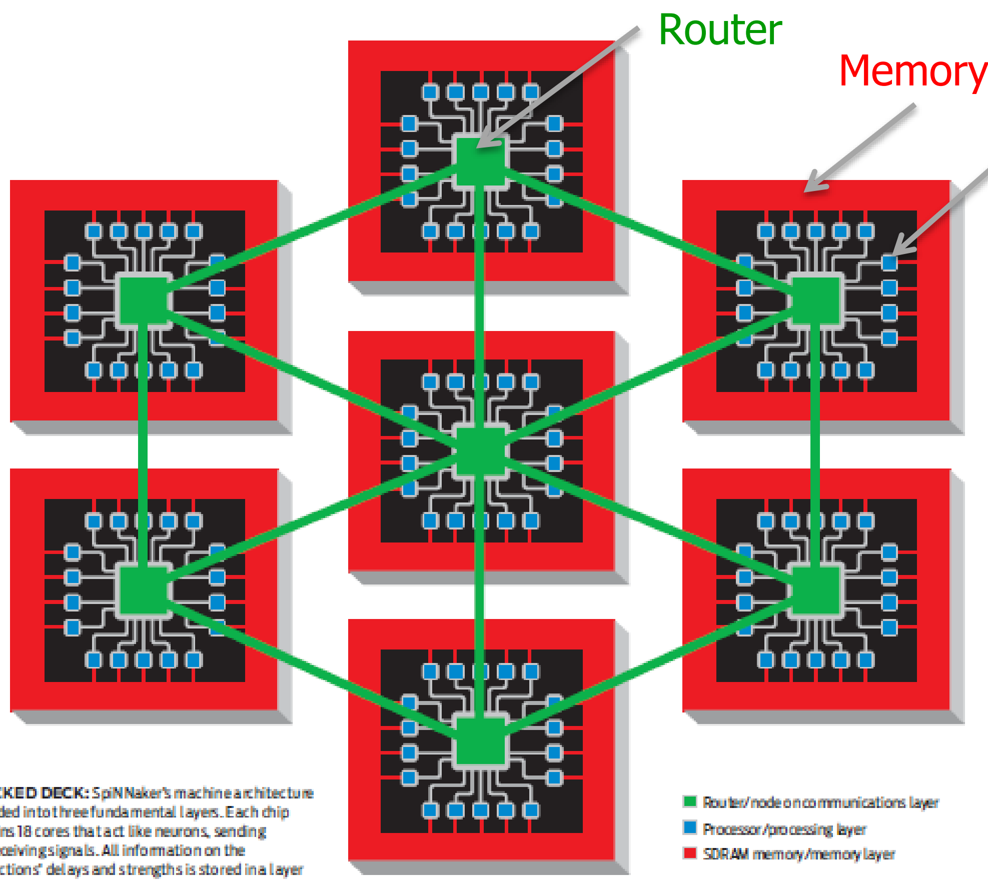
- A neuromorphic computer is a machine comprising many simple processors / memory structures (e.g. neurons and synapses) communicating using simple messages (e.g. spikes).
- Neuromorphic computers are one "pole" in a continuum of repurposable computing architectures



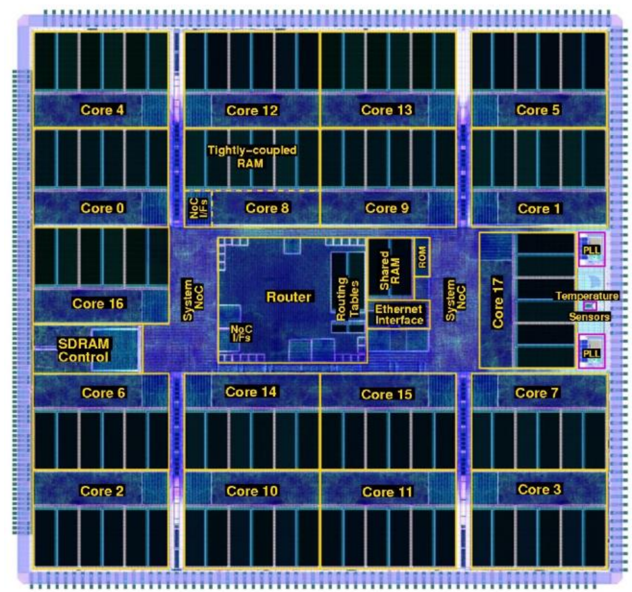
- Neuromorphic algorithms emphasize the *temporal* interaction among the processing and the memory.
  - Every message has a time stamp (explicit or implicit)
  - Computation is often largely event-driven

I think of neuromorphic computers as a kind of "dynamical" computer in which the algorithms involve complex spatio-temporal dynamics on the computing hardware.

Manchester University - SpiNNaker

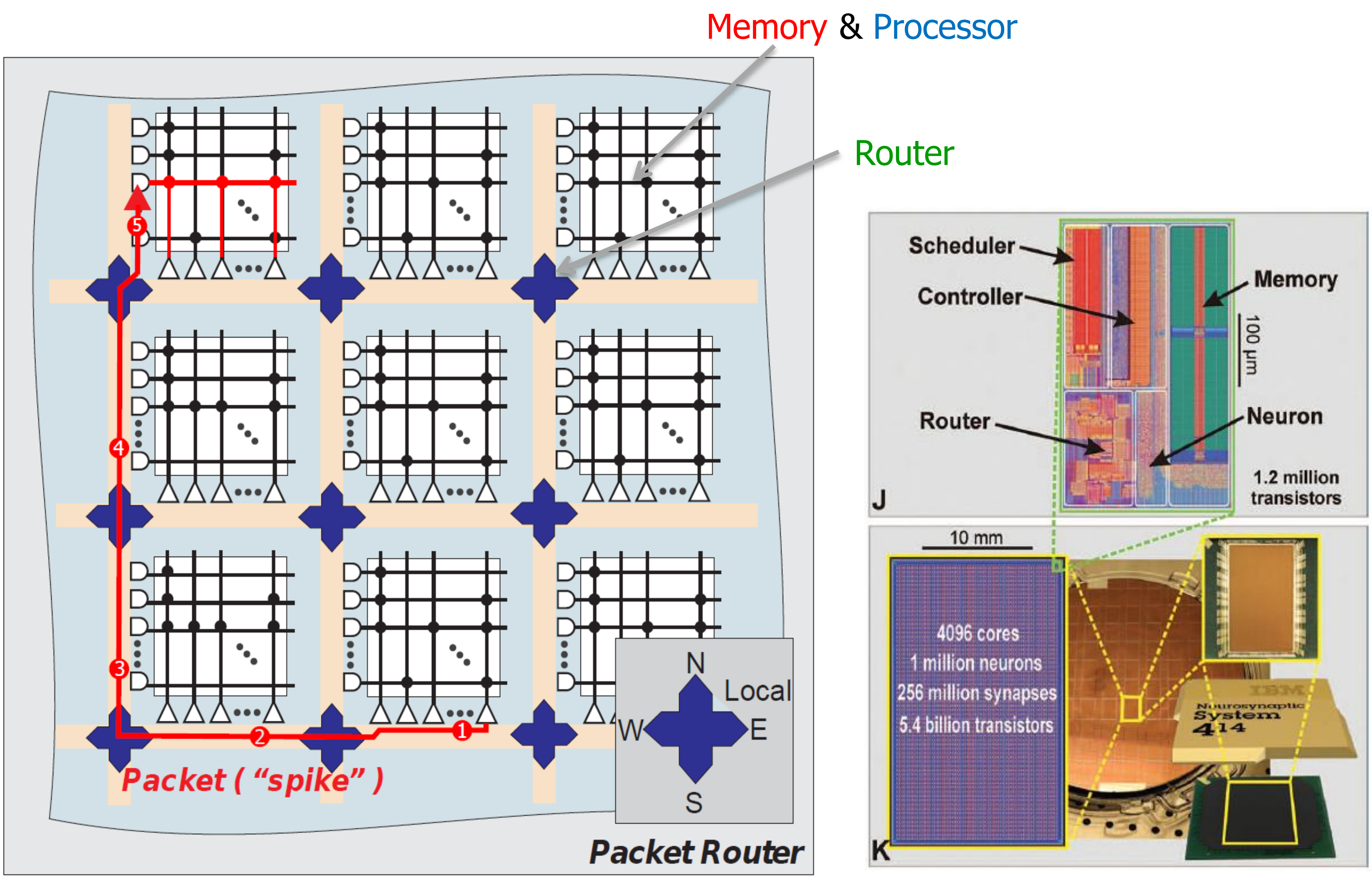


STACKED DECK: SpiNNaker's machine architecture is divided into three fundamental layers. Each chip contains 18 cores that act like neurons, sending and receiving signals. All information on the connections' delays and strengths is stored in a layer of synchronous dynamic RAM (SDRAM) on each chip, and all signals pass through a separate router layer.



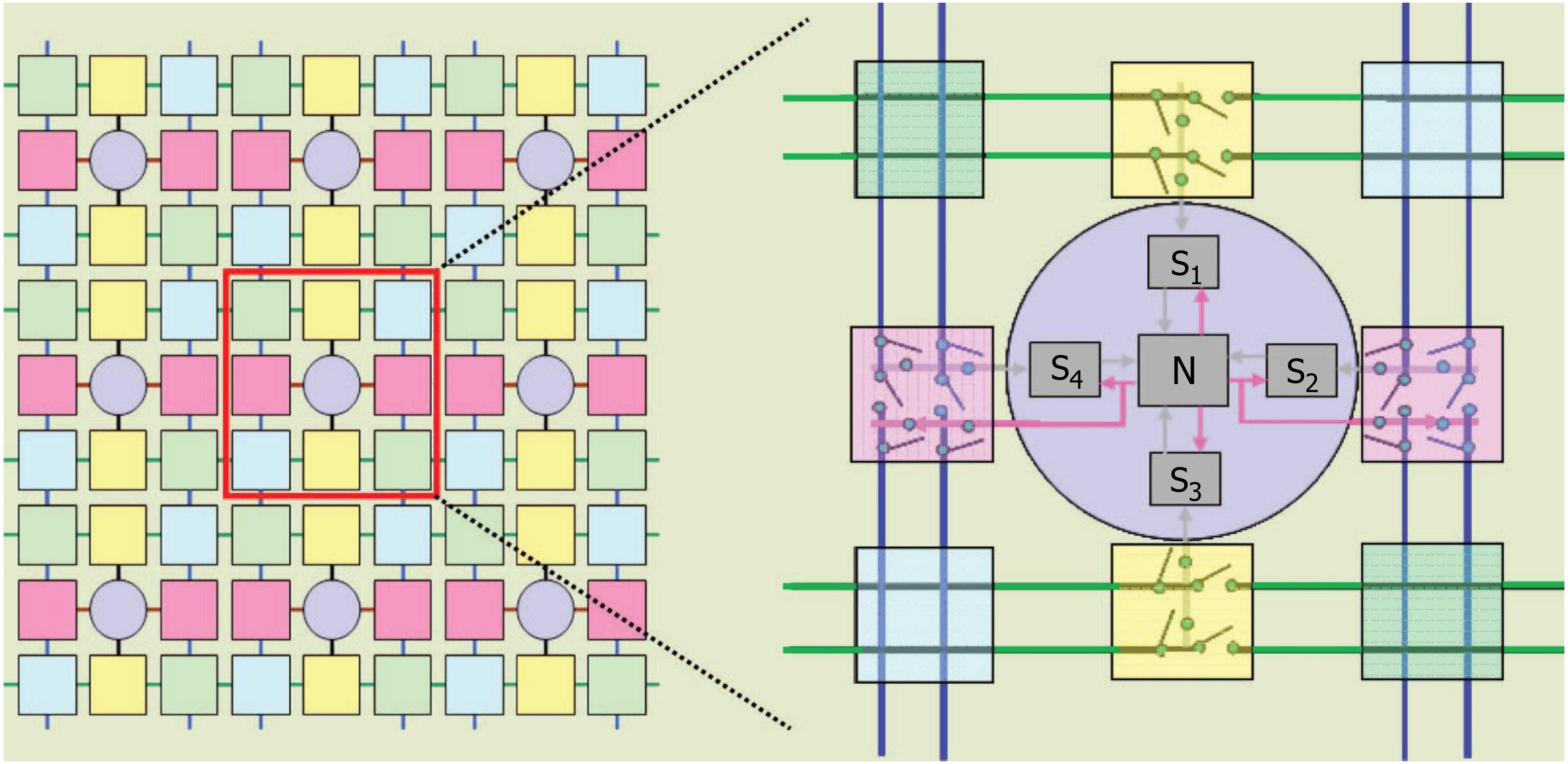
Steve Furber, "To Build a Brain" , *IEEE Spectrum*, August 2012

IBM – "True North"



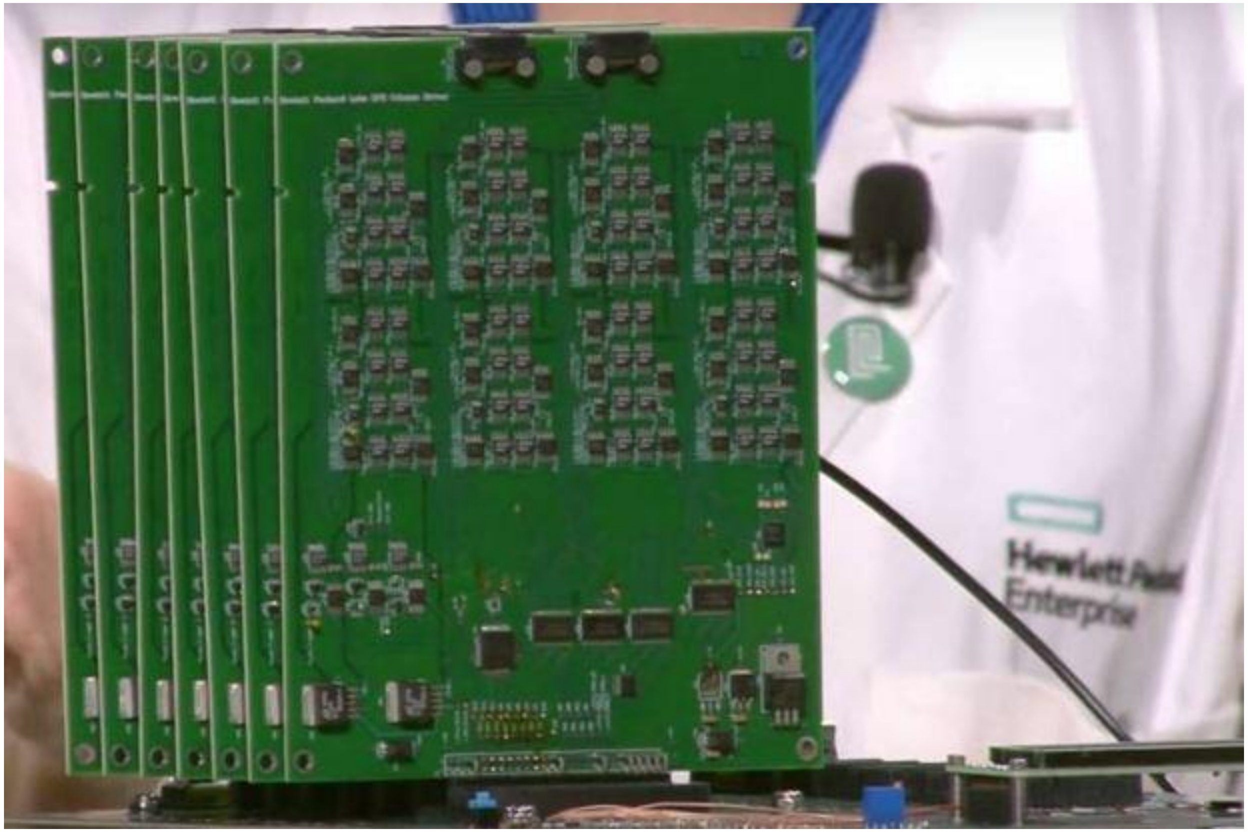
Paul A Merolla *et al.*, "A million spiking neuron integrated circuit with a scalable communication network and interface", *Science* **345** (2014)

HRL Labs – SyNAPSE Neuromorphic Architecture



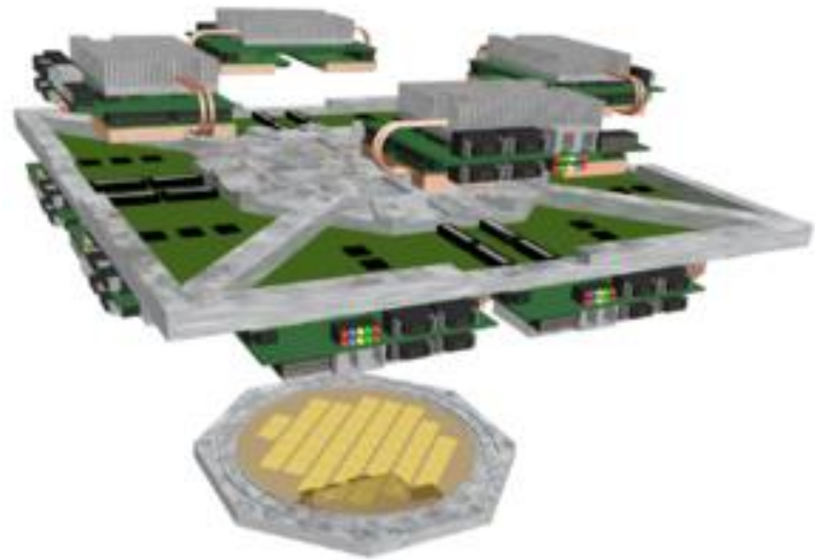
Narayan Srinivasa and Jose M. Cruz-Albrecht, “Neuromorphic Adaptive Plastic Scalable Electronics”, **IEEE PULSE,** JANUARY/FEBRUARY 2012

HP Enterprise - DPE



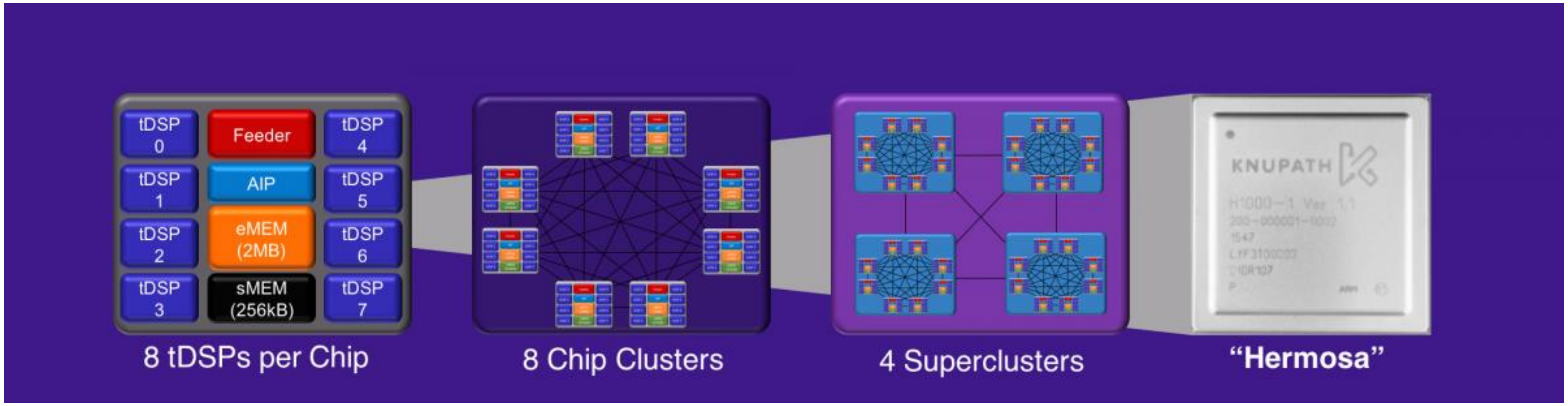
- “Dot Product Engine”
- Memristor memory / computation
- HPE Cognitive Computing Toolkit (“CogX”)

University of Heidelberg – BrainScales / HBP



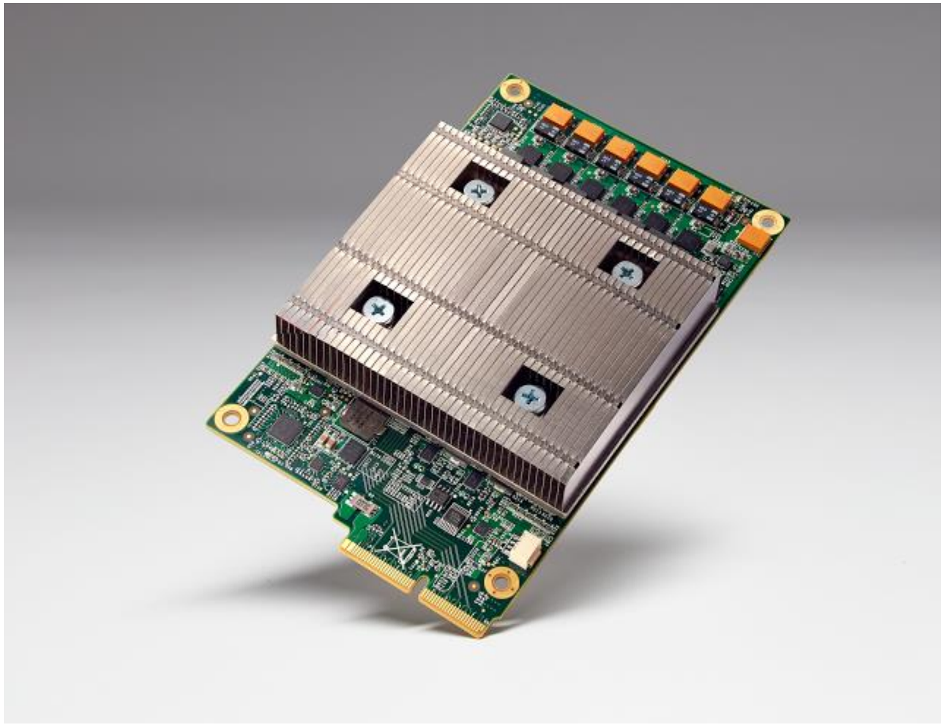
- Wafer scale neuromorphic architecture
- HBP - Neuromorphic Computing
- High-speed brain modeling

KnuEdge - KNUPATH



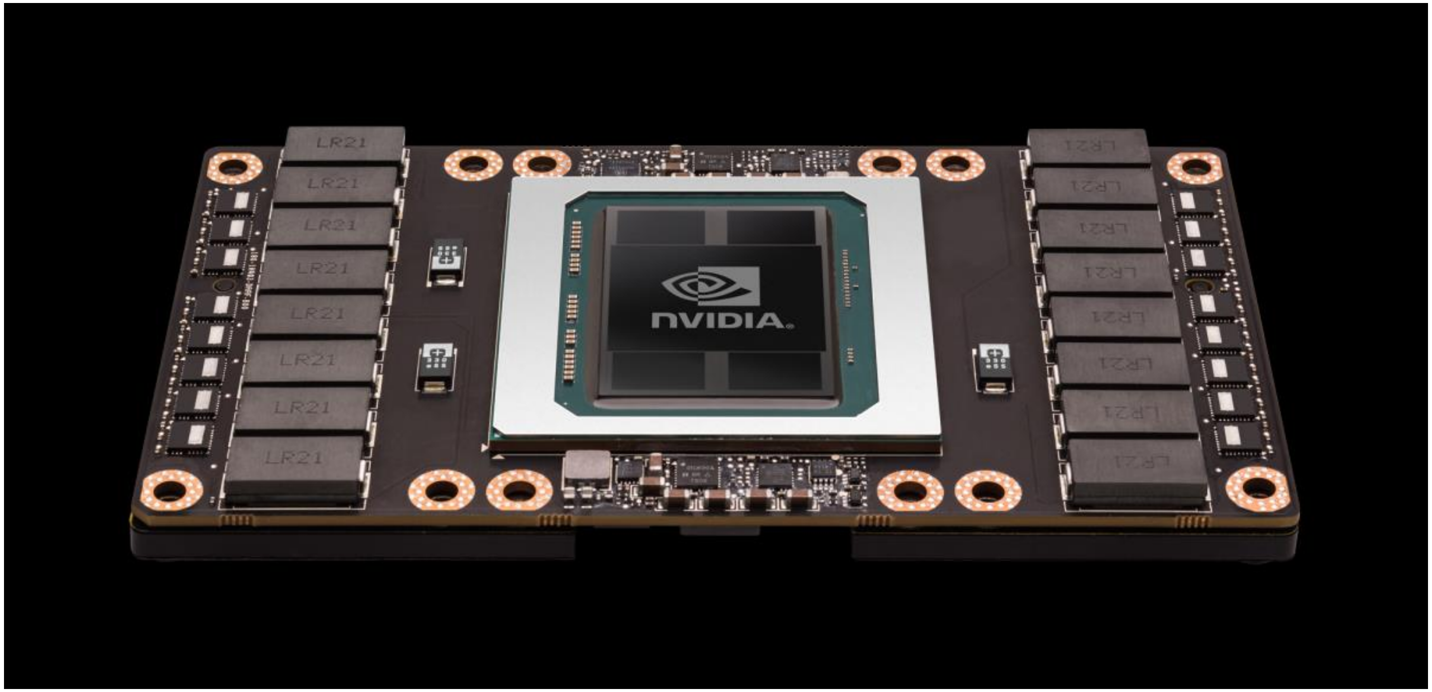
Hermosa chip with "LambdaFabric"

Google - TPU



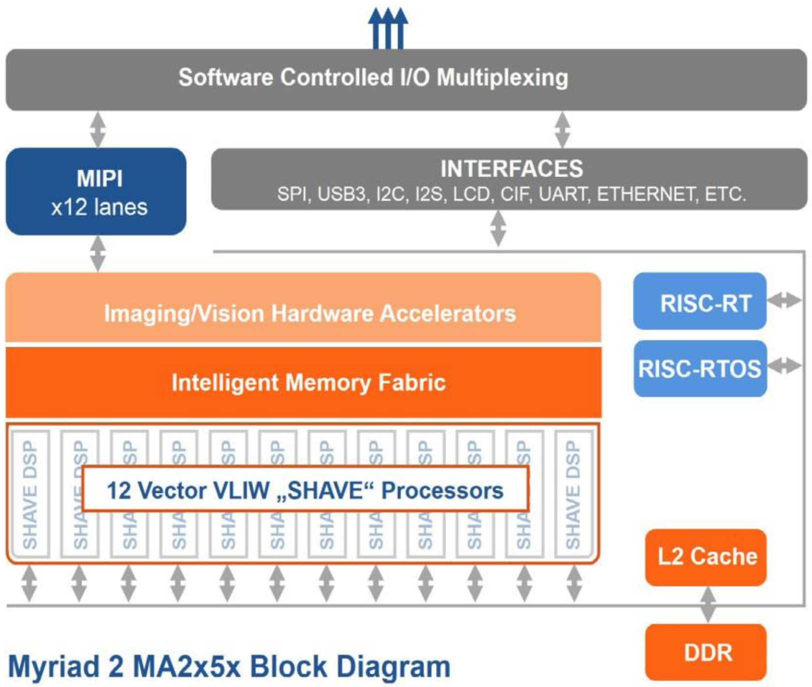
- “Tensor Processing Unit”
- Deep Learning Accelerator
- Runs TensorFlow

Nvidia - GPU



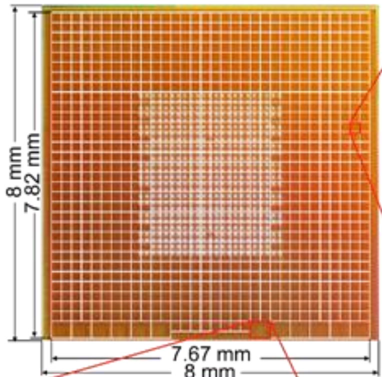
- NVIDIA Tesla P100
- Deep Learning acceleration
- cuDNN

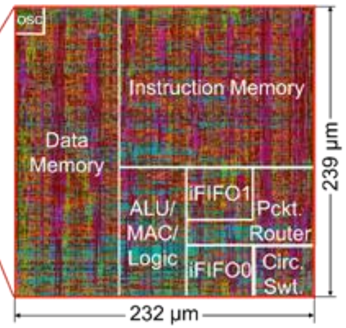
Movidius - VPU

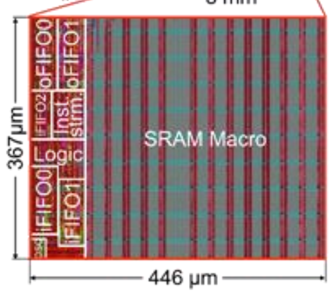


Fathom Neural Compute Stick

UC Davis – 1000 Processor Chip



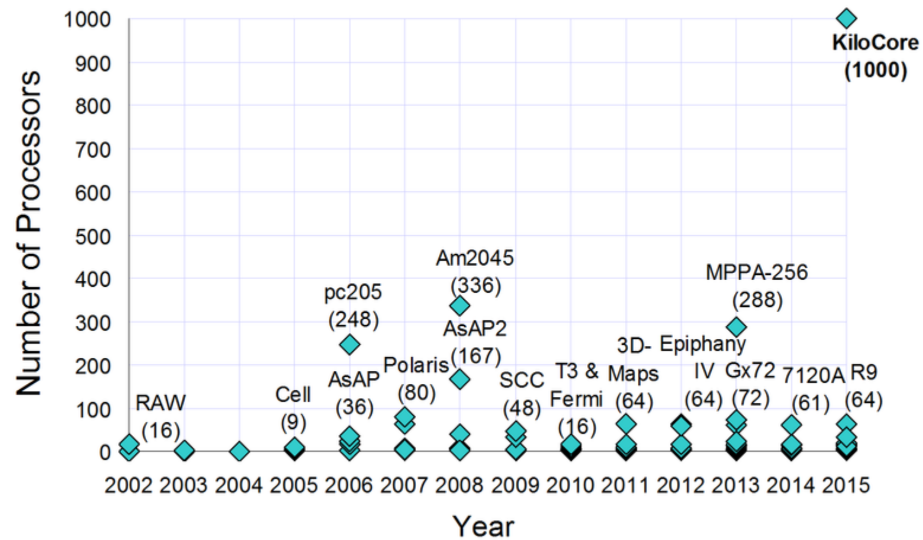




Chip	
Technology	32nm IBM PD-SOI CMOS
Num. Cores	1000
Num. Mems.	12
Num. Oscs.	2012
Die Area	64 mm²
Trans.	621 M
C4 Bumps	564 (162 I/O)
Package	676 Pad Flip-Chip BGA

Processor	
Tile Area	0.055 mm²
Transistors	575,000
Instruction Memory	128 x 40-bit
Data Memory	256 x 16-bit
iFIFO(0/1) Size	32 x 16-bit
Instruction Types	72

Independent Memory	
Tile Area	0.164 mm²
SRAM Macro Size	64 KB
(i/o)FIFO(0/1) Size	32 x 16-bit
iFIFO2 Size	16 x 2-bit



Where is Neuromorphic Computing Today?

- Expressions of the technology today
  - Goals and motivations are varied
  - Hardware prototypes are appearing regularly
  - Development tools are emerging
  - Existing algorithms are being ported to the new hardware
  - Applications and business models are uncertain
  - Substantial (but disconnected) activity across large tech companies, startups, government labs and universities

We have passed the "reasonability" and "feasibility" stages, have started the "development" stage, and can foresee an upcoming "utility" stage.

How "Neuromorphic" Are We?

- Deep learning algorithms are broken into "layers" and "steps", so although the units may be neuromorphic, the system dynamics are not.
- Most of the multi-core hardware is still synchronous and optimized for problems that are easily parallelized.
- The most common benchmarks consist of static datasets and classification tasks (instead behaviors in a dynamic, real-world environment).
- We don't have anything like general-purpose learning.
- The compute hardware does not operate close to any thermodynamic limit (as brains do).
- We need more memory per compute element.

Although we have made great progress and the field is rapidly evolving we still have a lot of room to improve. <u>We are not at the end of computing, we are at the beginning of a new paradigm.</u>

Prospective - Technology Landscape

Technology Landscape

Dynamic, Online, Real World

Program ← → **Learn**

- Sensing
- Display
- Wireless communication & internet
- Computing at the edge

Mobile Phones

- IoT
- Robotics
- Industrial Internet
- Self-driving Cars
- Smart Grid
- Secure Autonomous Networks
- Real-time Data-to-Decision

Intelligent Cyber-Physical Systems

- Personal Computing
- Wired Internet

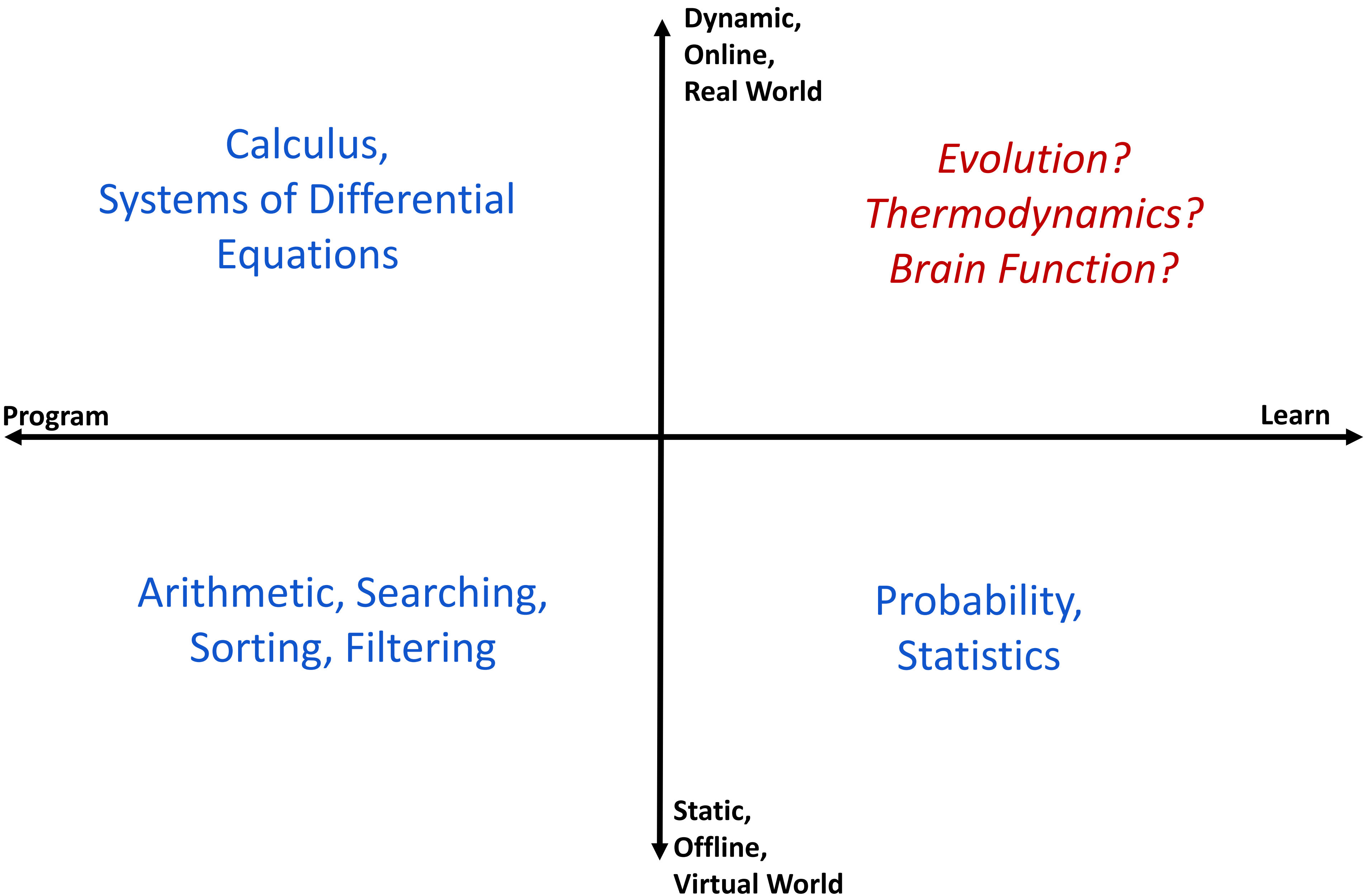
Desktop / Workstation

- Data Integration
- Large scale storage
- Large scale computing

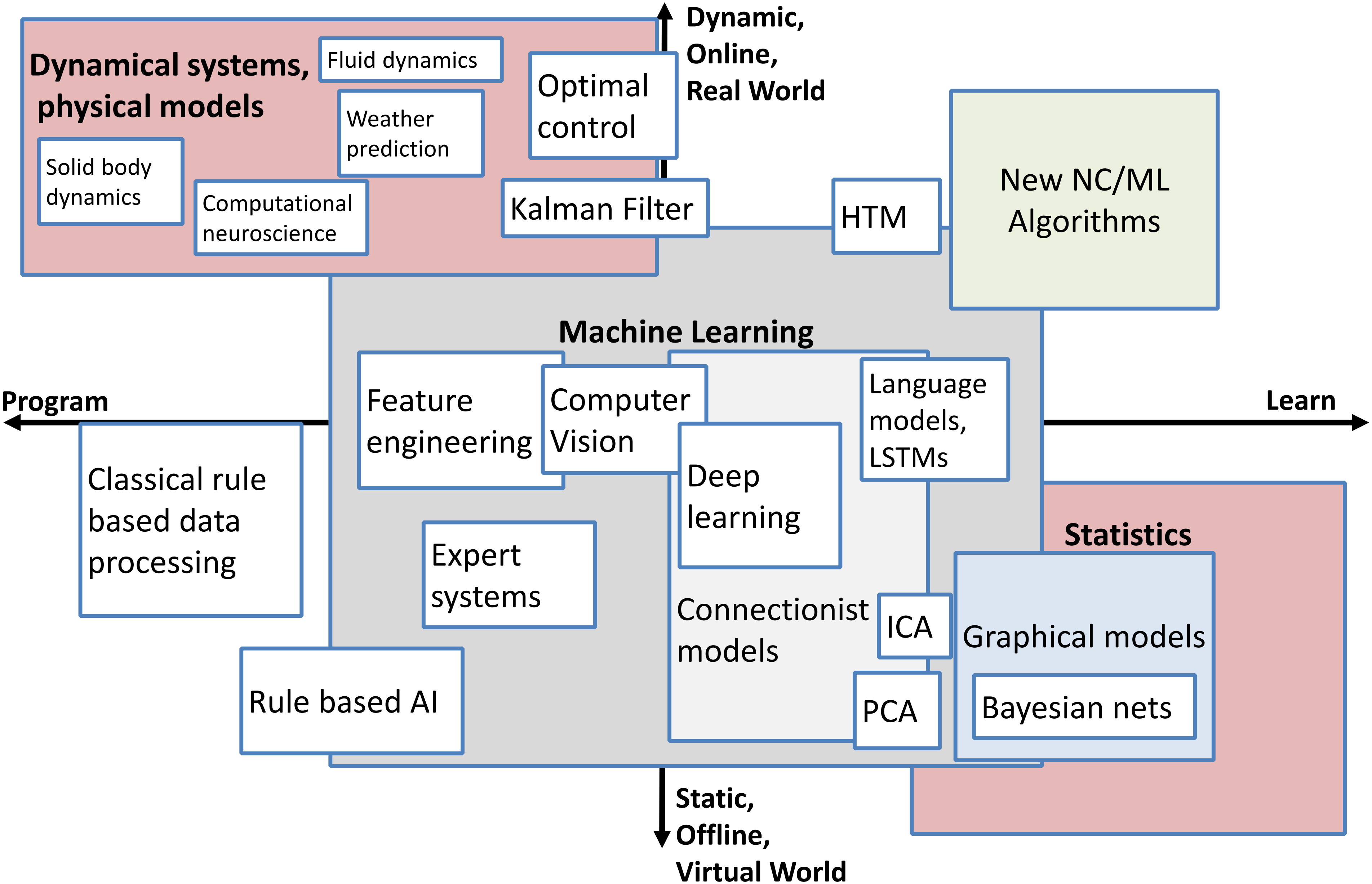
Data Center / Cloud

Static, Offline, Virtual World

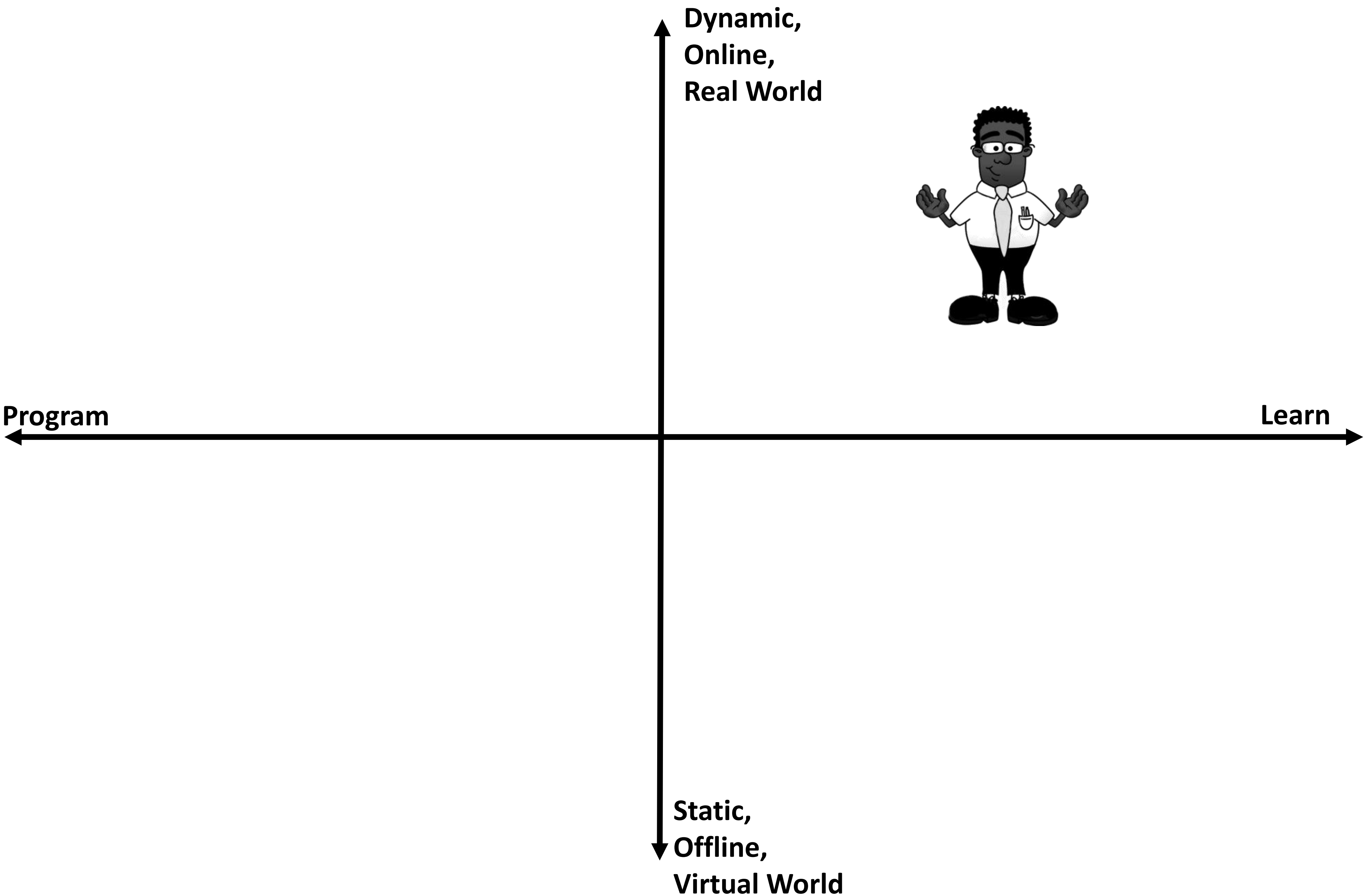
Conceptual Landscape



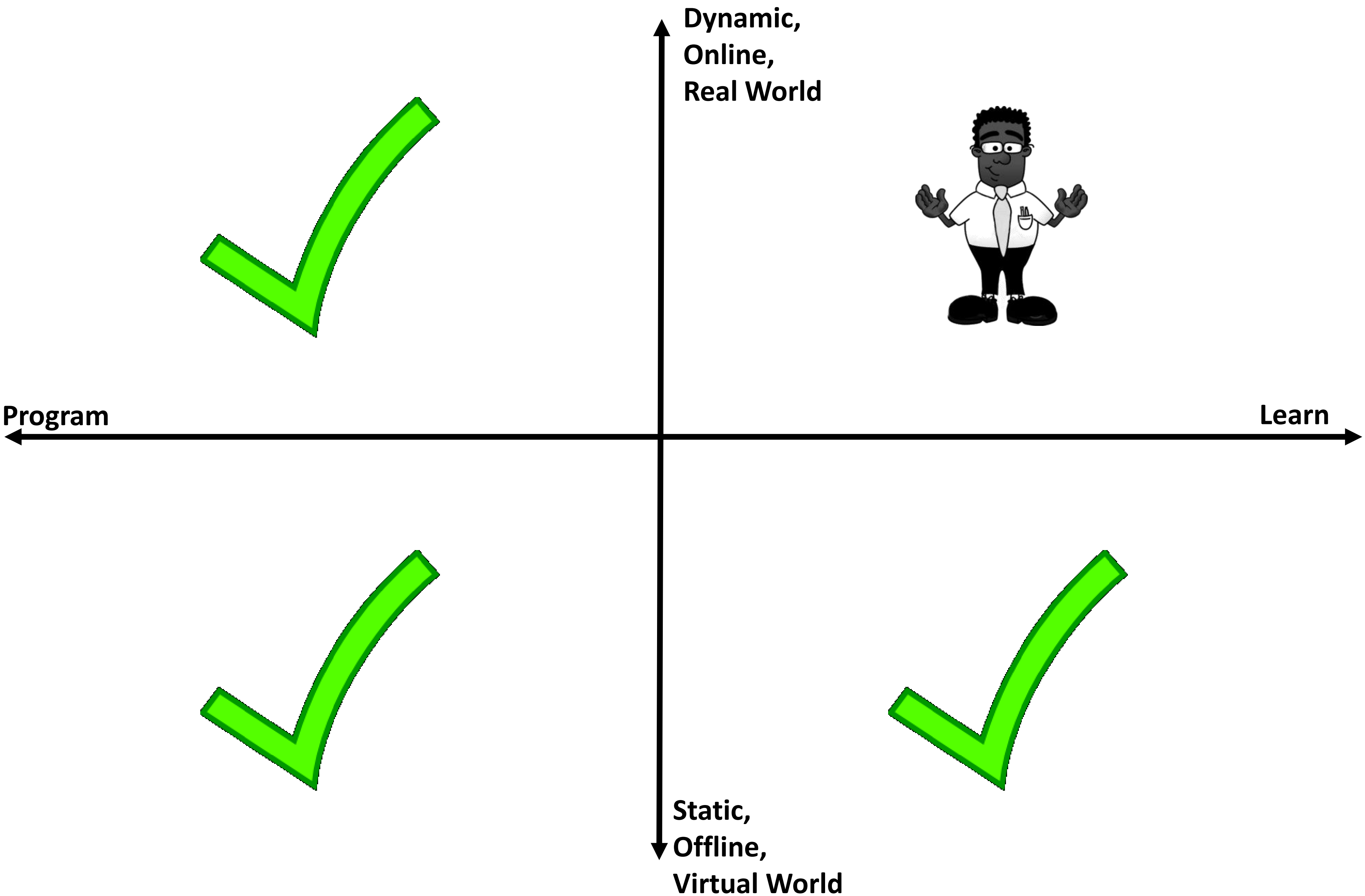
Algorithm Landscape



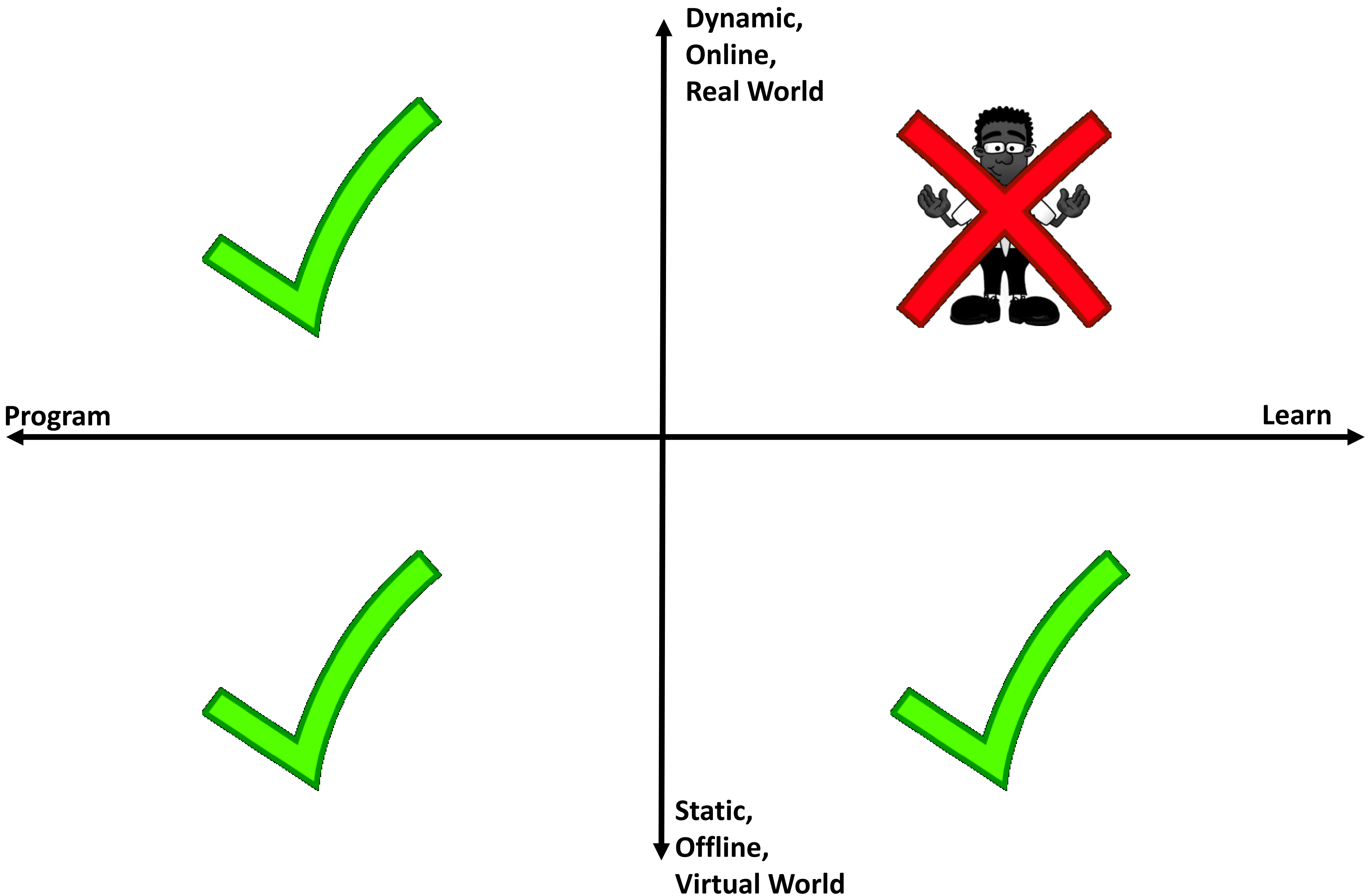
Programming Landscape



Programming Landscape



Programming Landscape



The Paradox of Programming a Neuromorphic Computer

- Although we have large brains we cannot (yet) program a neuromorphic computer very well.
  - Our high level communication and thinking (language) seems to be composed of a large long-term memory and a small symbolic processing capability and this is what we use to design and build a (Von Neumann) computer
- The paradox is expressed in the many dualism of Von Neumann computation that become muddled / problematic in neuromorphic computing
  - Hardware vs. Software
  - Logic vs. Memory
  - Computation vs. Communication
  - Program vs. Data
- To me it suggests that
  - We need to figure out how to write a new class of algorithms
  - We are missing / unaware of some important basic concepts
  - If learning is the answer, what is learning?

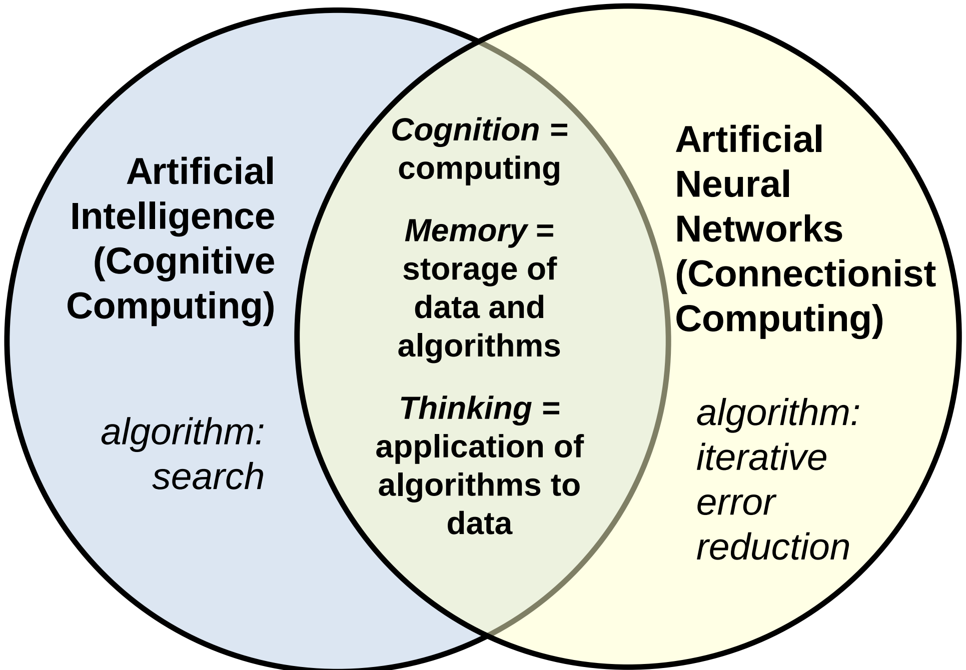
Prospective – Framing the Opportunity

Traditional neuromorphic / cognitive computing proposition

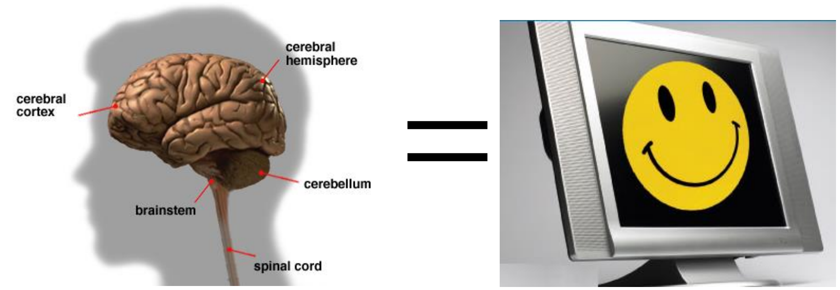
Build computers that learn and generalize in a broad variety of tasks, much as human brains are able to do, in order to employ them in applications that require (too much) human effort.

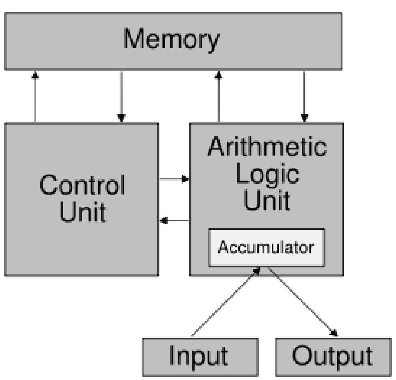
- This idea is at least 40 years old, yet we still don’t have these kinds of computers.
- We have become disillusioned with these ideas in the past because the proposition was not fulfilled (AI and neural net “winters”)
- The proposition is (very) popular again because
  - Maturation of the computing industry
  - The successful application of some machine learning techniques
  - Interest and research on the brain

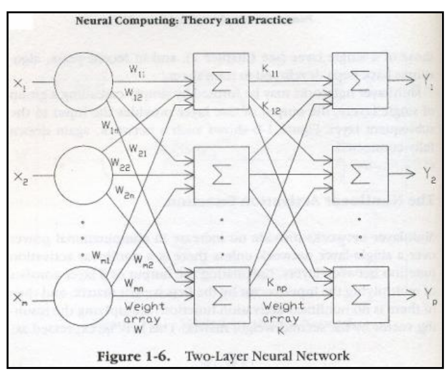
Neuromorphic / cognitive computing philosophy



Cognitive computing views the brain as a computer and thinking as the execution of algorithms.







- Biological memory corresponds to a container holding data and algorithms. Learning fills the container with input-output rules defined on discrete (AI) or continuous (ANN) variables.
- Algorithms create input-output mappings using rules or weights stored in memory.
- AI focuses on search algorithms to select “production” rules.
- ANN focuses on iterative error reduction algorithms to determine “weights” yielding the desired input-output relationships.
- Algorithms are created by humans.

What about Machine Learning?

- Machine learning refers to a collection of computational methods / algorithms that refine (typically) many parameters in order to associate an input dataset with a desired output.
- The algorithms optimize an internal objective function that is coupled to input datasets and (labeled) output associations.
- Algorithms have narrow domain of application and are typically tied to the datasets / benchmarks that they seek to represent.
- **A machine learning algorithm is not a brain.** Humans are required to write the algorithms, provide the input datasets, and the output objectives.

Machine learning is a collection of powerful computational techniques for discovering statistical regularities in well-defined input datasets and associating them with well-defined outputs.

Building Intelligent Systems

Practical Computation

Computational Complexity

Electronics Technology

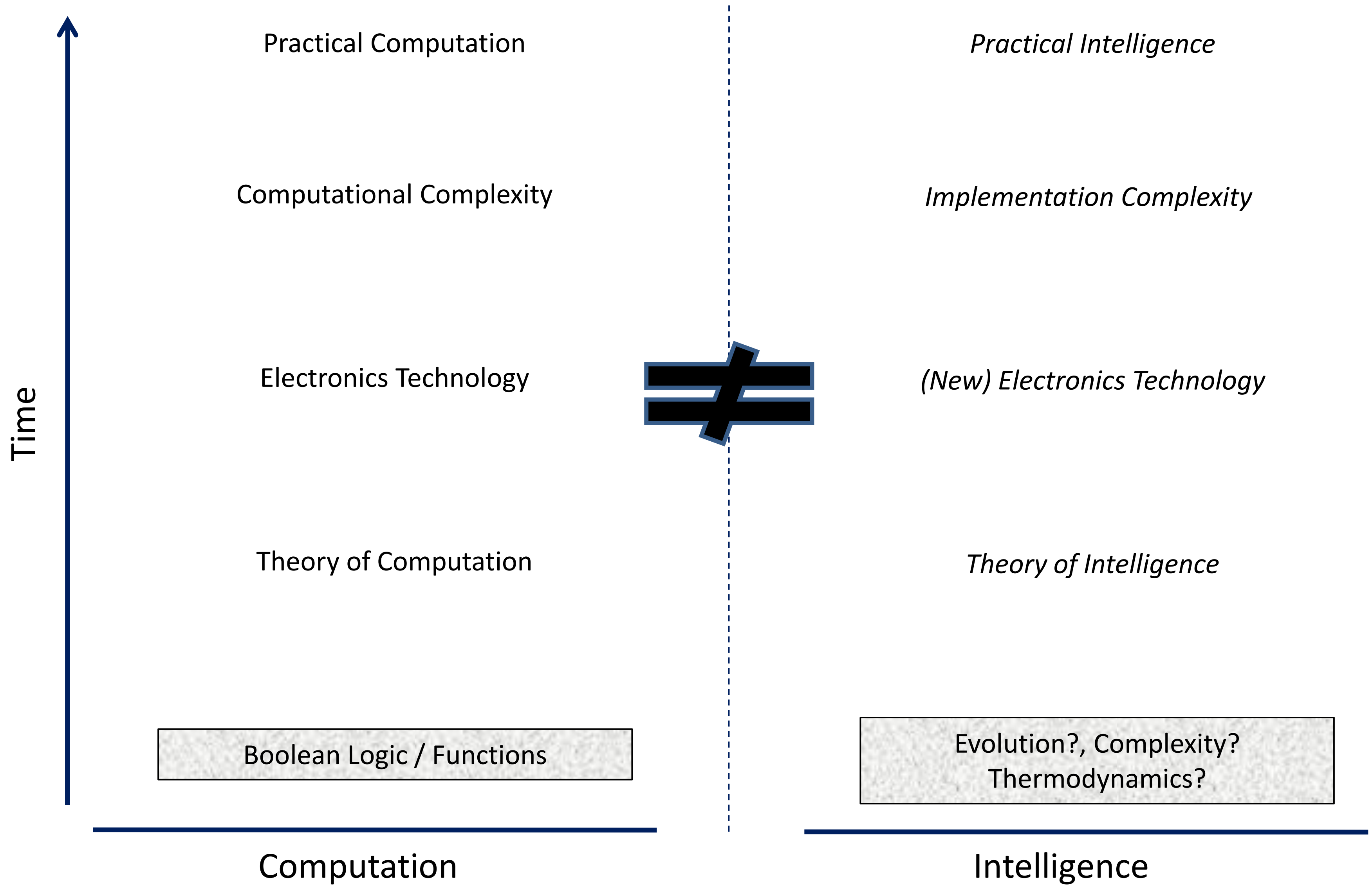
Theory of Computation

Boolean Logic / Functions

Time

Computation

Building Intelligent Systems



Building Intelligent Systems

Practical Intelligence
Practical Computation

Implementation Complexity
Computational Complexity

(New) Electronics Technology
Electronics Technology

Theory of Intelligence
Theory of Computation

Evolution?, Complexity?
Thermodynamics?

Physics is missing

Time

Intelligence & Computation

Getting it Straight – Understanding Neuromorphic Computing

- A neuromorphic computer is another kind of repurposable computing platform like a CPU, GPU, FPGA, etc.
- A neuromorphic computer will be more / less efficient than another computing architecture depending on the algorithm
- Neuromorphic computers may be good choices for implementing some machine learning algorithms, but these should not be confused with brains
- **A neuromorphic computer is not a brain**. Although if we figure out how to create the intelligence that we associate with brains on a computer, a neuromorphic computer would likely be an efficient option.

Getting it Straight - Understanding Intelligence

- In the early days of computing, thinking in terms of basic physical and philosophical ideas were common.
  - *In fact, there are numerous indications to make us believe that this new system of formal logic will move closer to another discipline which has been little linked in the past with logic. This is thermodynamics, primarily in the form it was received from Boltzmann, and is that part of theoretical physics which comes nearest in some of its aspects to manipulating and measuring information.* – John Von Neumann
  - *To suppose universal laws of nature capable of being apprehended by the mind and yet having no reason for their special forms, but standing inexplicable and irrational, is hardly a justifiable position. Uniformities are precisely the sort of facts that need to be accounted for. Law is par excellence the thing that wants a reason. Now the only possible way of accounting for the laws of nature, and for uniformity in general, is to suppose them results of evolution.* - Charles Sanders Peirce
  - *The extraordinary integration and interdependence of the universe over massive spatial and temporal scale is a consequence of evolution from a common starting point and organizing principle. We are part of this universe and our own intelligence is one manifestation of this principle.* - TLH
- Understanding intelligence implies understanding even broader questions.
- Today we lack the conceptual foundations to be proficient at building intelligent systems.

Revised neuromorphic / cognitive computing proposition

- *Build computers using a large number of highly-distributed computational elements, embedded memory, and a reconfigurable messaging network in order to efficiently process algorithms having complex spatio-temporal dynamics, large data flow, and many adaptable parameters.*
- *In order to proficiently build intelligent systems, create an understanding of intelligence derived from basic principles and translate this understanding into all aspects of neuromorphic system development.*

Prospective – Goals for the Future

Potential Application Domains of Neuromorphic Computing

- Automobiles
- Phones, computers, tablets, etc.
- Large scale commercial, scientific, intelligence data analysis
- Massive, distributed sensor networks
- Commercial, consumer, industrial robotics
- Commercial, consumer, industrial IoT (21B devices projected by 2020)
- Smart grids / cities / buildings / factories
- Cyber security
- Cyber warfare
- Autonomous defense systems and networks (UAV, UGV, UUV…)
- *Everything with lots of data…*

The application domain is <u>enormous</u> but also poorly realized because the necessary technologies do not yet exist.

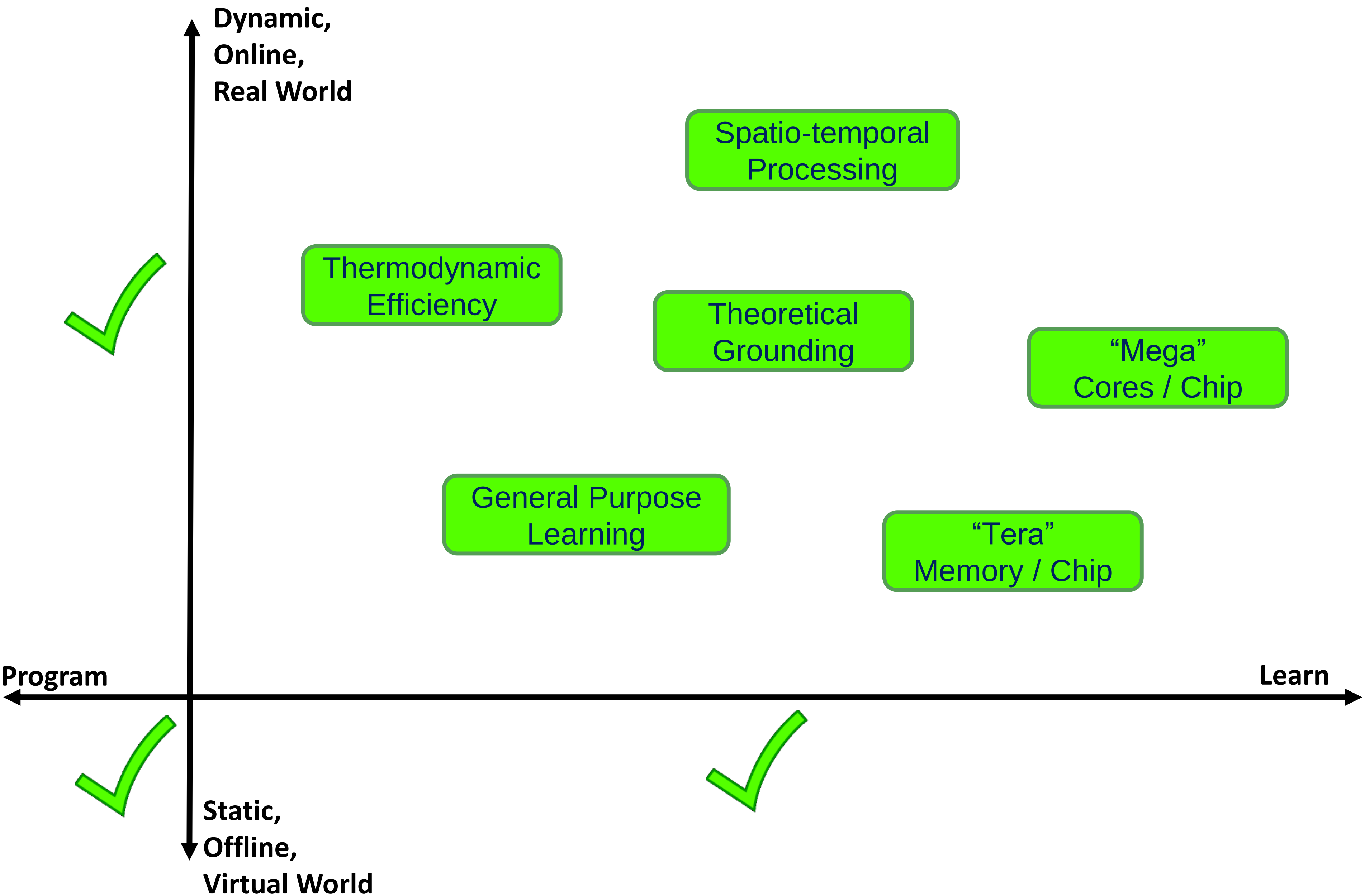
Challenges / goals for the future

- Can we build a simulator that supports different neuromorphic architectures?
- Can we build tools to map algorithms to those architectures?
- Can we estimate the architecture / algorithm performance in hardware?
- Can we create a suite of benchmarks to test the relative strengths and weaknesses a neuromorphic computing approach?
- Can we build high density memories local to the processing elements in/on state-of-the-art CMOS?
- Can we move beyond our current step-at-a-time thinking to programming?
- Can we create a general purpose learning methodology?
- Can we develop the conceptual foundations of intelligence?
- Can we leverage industry, academic and government laboratory efforts?
- Can we make neuromorphic computing a strategic, national priority?
- Can we invest in both short term opportunities and long term objectives?

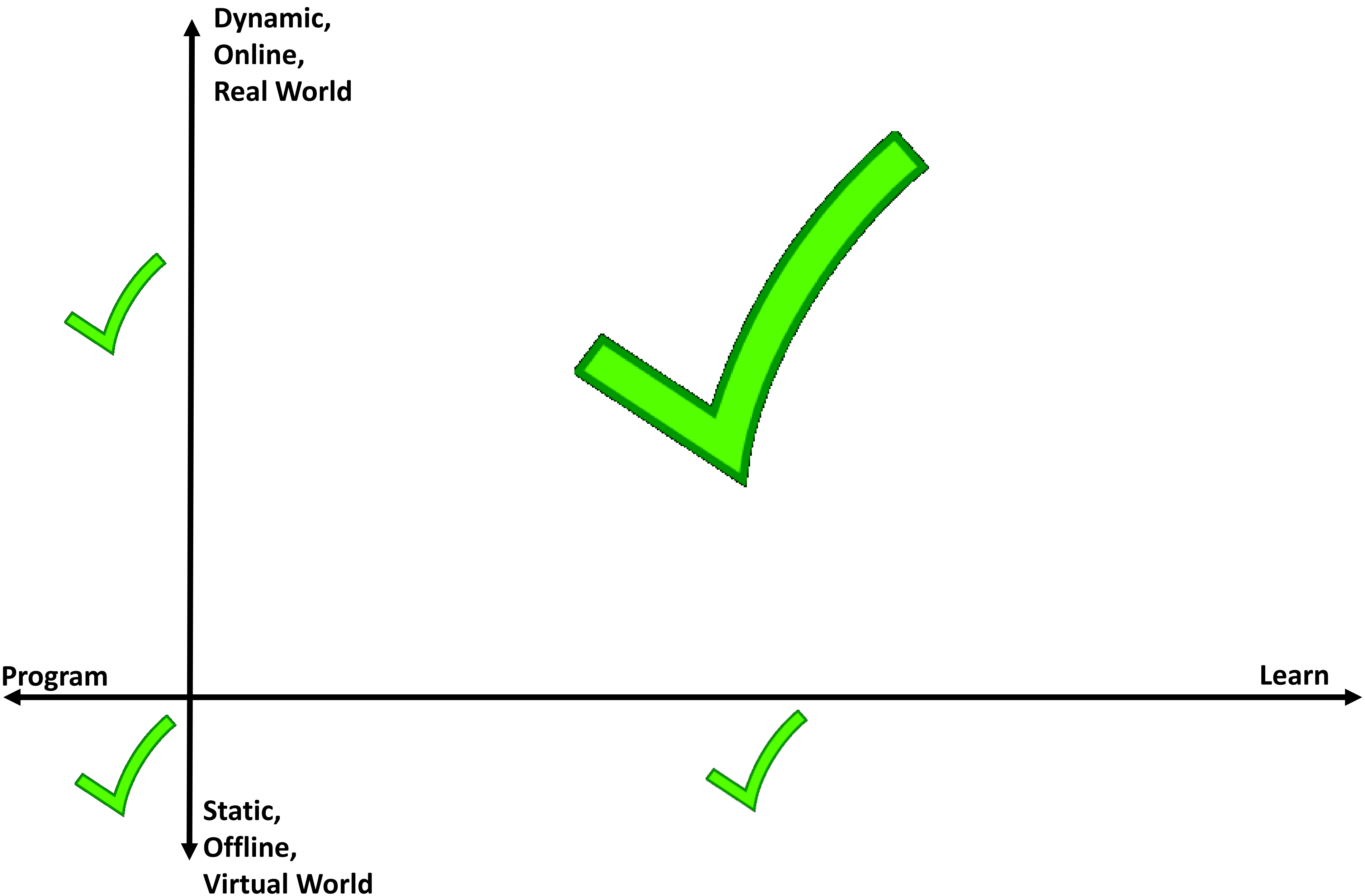
Why I am Bullish

- The maturation of the current computing technology invites disruption by new ideas
- The applications of the future require a neuromorphic computing solution.
- Neuromorphic computing and the motivation to build intelligent systems from them will not only create massive economic and societal benefit, but will also create a new understanding of ourselves and, thereby, transform all human endeavor and experience.

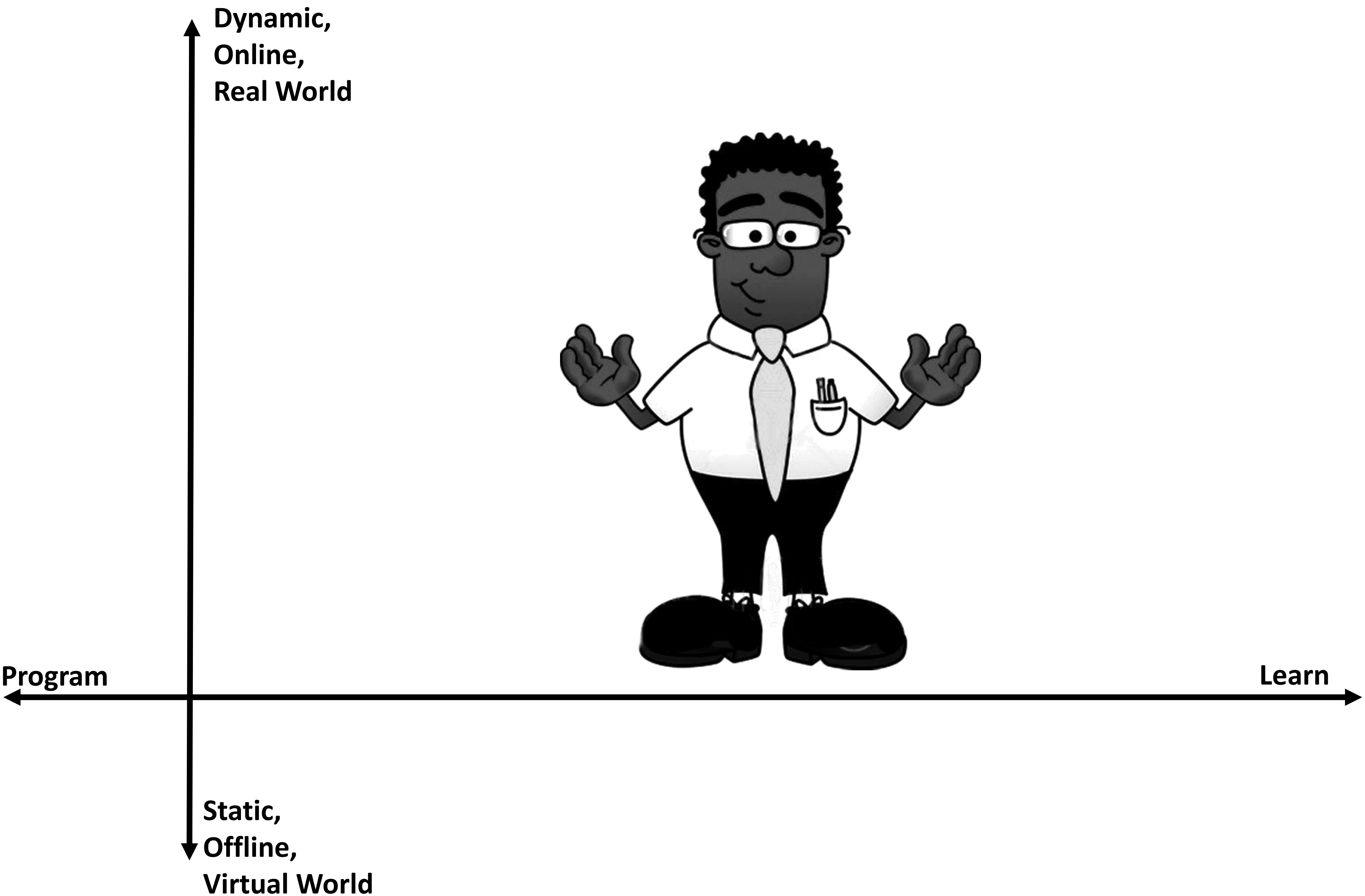
Future Landscape



Future Landscape



Future Landscape



Why the government should invest

- The end of Moore’s Law portends a paradigm shift offering both disruption and opportunity
- The massive, on-going accumulation of data everywhere is an untapped source of wealth and well-being for the nation
- The need for on-line, adaptive, autonomous systems in conventional and cyber warfare
- The threat of large, nation-state adversaries gaining prominent capabilities – “Sputnik”
- The ubiquitous availability of computing resource and training for those interested in developing neuromorphic computing / machine learning technology gives many the opportunity to disrupt
- The likelihood of breakthroughs in fundamental science driven by the quest for neuromorphic computing and its ultimate realization
- The commercial sector will not invest in the early stages of a paradigm shifting technology
  - E.g. deep learning did not originate in Silicon Valley with Venture funding, it is the product of decades of government funded R&D (as is virtually every other game-changing computer technology).
  - Silicon Valley exists because the US DoD and NASA funded the development and bought the products of the nascent semiconductor industry in the 1950s.
- Government applications are different than commercial applications, so many government needs will not be met if they rely on technology derived from commercial products
- The long-term economic return of government investment in neuromorphic computing will likely dwarf other investments that the government might make
- The government’s long history of successful investment in computing technology (probably the most valuable investment in history) is a proven case study that is relevant to the opportunity in neuromorphic computing