



LOIHI DEEP DIVE

Architecture, SDK, Examples

Neuromorphic Computing Lab | Intel Labs

March 29, 2019

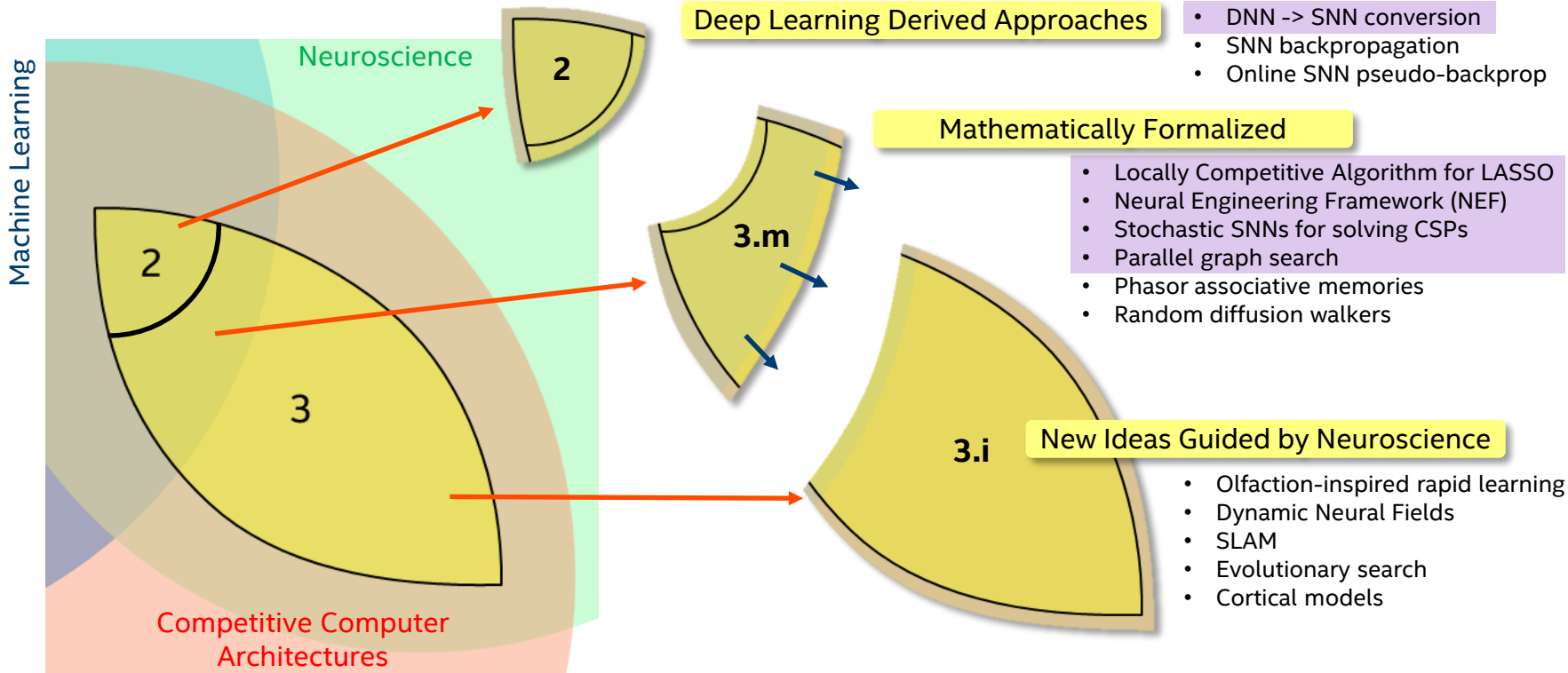
Neuro-Inspired Computational Elements, SUNY Polytechnic Institute

SCHEDULE FOR THE DAY

Morning Session	
9:15 – 10:15	Loihi Architecture Overview
10:15 – 10:25	Break / Hardware Q&A
10:25 – 10:55	NxSDK Architecture
11:00 – 11:35	NxNet Intro <ul style="list-style-type: none">• Add compartments/connections (code, basic behavior)• STDP Learning and eligibility traces• Kapoho Bay DVS Demo• Multi-compartment neurons – <i>time permitting</i>
11:35 – 11:45	Software Q&A

Afternoon Session	
1:00 – 1:10	NxSDK Overview (quick version)
1:10 – 2:05	Algorithmic Demos with NxNet <ul style="list-style-type: none">• Single-Layer Image Classification• Solving LASSO w/ Spiking LCA• Constraint satisfaction
2:05 – 2:35	Algorithmic Demos with Nengo <ul style="list-style-type: none">• Nonlinear oscillator• Learning w/ Prescribed Error Sensitivity• MNIST classification with Nengo DL• Keyword spotting with Nengo DL
2:35 – 2:50	Graph Search and Multi-Chip Scaling
2:50 – 3:00	Closing / Q&A

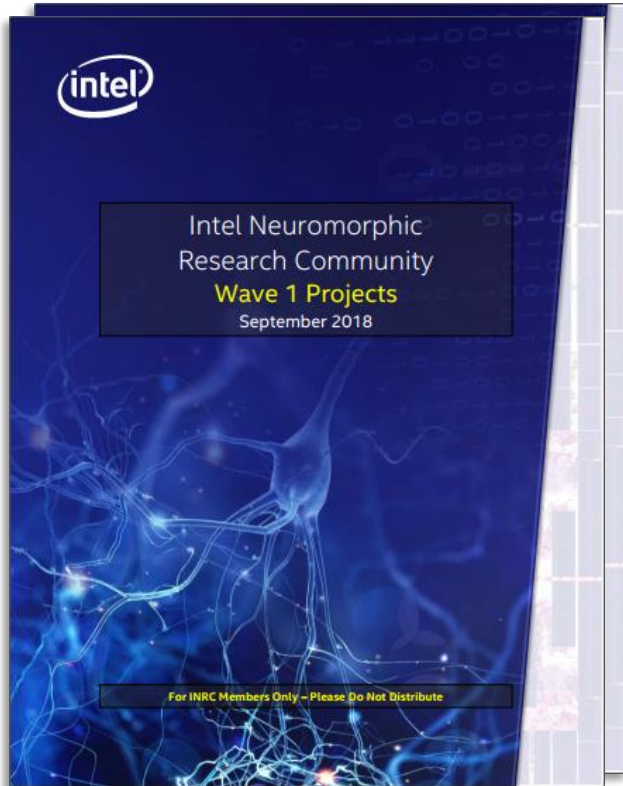
SNN Algorithms Discovery and Development



Intel's Objectives for INRC

- 1) Accelerate** research in neuromorphic computing
By stimulating *algorithms* and *applications* research focusing on Loihi architecture
- 2) Quantify** the value of neuromorphic computing today
Discipline of a quantified approach is critical for progress and mainstream adoption
- 3) Inform Loihi's architectural development**
Algorithms & application findings provide insights for future silicon revisions
- 4) Build an ecosystem** that can provide a market for neuromorphic chips
Intel hopes to sell chips, eventually... we need customers and a broad user base

Join the Community



Contents

Research Vector 1: Theory.....	
Neuromorphic Complexity Analysis Johan Kweithout, Radboud University.....	
Research Vector 2: Algorithms.....	
Reinforcement Learning by Advanced Spiking Neuronal Networks Maxim Bazhenov, UC San Diego.....	
Olfactory System-Inspired SNN Algorithm for Signal Restoration and Identification Thomas Cleland, Cornell University.....	
Olfactory System-Inspired SNN Preprocessing Algorithms for Signal Conditioning Thomas Cleland, Cornell University.....	
Principled Development of Drastically Improved Architectures and Algorithms for Computing and Learning with Neurons on Loihi Wolfgang Maass, TU Graz.....	
Principled Derivation of Spiking Neural Networks that Learn from Data Gangjo Parkhavan, Harvard Dmitri B. Chklovskii, RWU.....	
Real-time Error-Backpropagation for Deep Cortical Microcircuits in Spiking Neuromorphic Systems Walter Senn, University of Bern.....	
Information-Theoretic Learning of Probabilistic Spiking Neural Networks Oswaldo Simeone, Bipin Rajendran, Andre Gruning, Michael Schumaker.....	
A Structured Approach to the Design of Algorithms for Loihi Friedrich Sommer, Bruno Olshausen, Pentti Kanerva, E Paxon Frady University of California Berkeley.....	
Cognitive Agents for Autonomous Robots Tarek Taha, University of Dayton Kerry Hill, Air Force Research Laboratory.....	33
Energy-efficient, Adaptive Control of Complex Sequences of Robotic Movements Christian Tetzlaff, University of Göttingen.....	35
Research Vector 3: Applications.....	38
Hierarchical Sparse Coding for Robustness against Adversarial Attack Edward Kim, Villanova University Garrett Kenyon, New Mexico Consortium.....	39
Loihi Simulator Backend for the Neurorobotics Platform Alois Knoll, Florian Walter, Mahmoud Aki TU Munich.....	42
Spiking Compliant Robot Control Alois Knoll, Florian Walter, Mahmoud Aki TU Munich.....	44
A Biologically Constrained Spiking Neural-Astrocytic Network Learns Action Habits for Autonomous Mobile Robots Constantine Michitizo, Rutgers University.....	46
Event-Driven Malware Detection Matilda Rhoads, Cardiff University Richard French, Airbus.....	52
Research Vector 4: Programming Models.....	54
Using the TENNLab Software Stack to Explore and Demonstrate Capabilities of Loihi James S. Plank, University of Tennessee.....	55
Research Vector 5: Event-Driven Interfaces.....	58
Interfacing ATIS to Loihi and Implementation of orthogonal transforms for gray events Siohio Ieng, Sorbonne Université Ryad Benosman, University of Pittsburgh.....	59
Neural Control Neuromorphic Adaptive Feedback Motor Control for Compliant Actuators Yulia Sandamirskaya, Institute of Neuroinformatics, University and ETH Zurich.....	62

RV2.27_Sommer

A Structured Approach to the Design of Algorithms for Loihi

Friedrich Sommer, Bruno Olshausen, Pentti Kanerva, E Paxon Frady
University of California Berkeley

ABSTRACT

The goal of this project is to leverage vector-symbolic architectures/hyperdimensional computing to build a structured and transparent framework for developing spike-based algorithms for the Loihi platform. Based on our novel model of computation with spiking neurons, we propose to develop a framework with two description levels to build algorithms on the Loihi platform. The computational level describes algorithms in terms of operations in high-dimensional representational spaces, like in vector-symbolic connectionist models of cognitive reasoning (Plate, 1995; Kanerva, 1996). The mechanistic level describes how these operations and representations are mapped onto hardware. To test and showcase the framework, we will use it for developing efficient models of associative memories and a hippocampus-inspired navigation system for autonomous, visually-guided spatial navigation to be embedded on mobile platforms.

DESCRIPTION

In this project, we will develop a modular and general design framework for algorithms that run on the neuromorphic hardware Loihi (Davies, 2018). The framework will leverage models that have been proposed in the connectionist literature to describe cognitive brain

The first aim is to implement associative memory algorithms that can make efficient use of synaptic memory to store data and access the data in a content-addressable fashion, even in absence of exact matches. Associative memory can be used for denoising and

Available to members:

- Access to member website
- Project documentation
- Access to GitHub site
- Participation in INRC workshops

INRC Engagement Process

- 1) Email **inrc_interest@intel.com**
We'll send you our RFP and project proposal template
- 2) Submit a **project proposal**
Tell us what you want to investigate and accomplish with Loihi
- 3) Execute the **INRC participation agreement**
Requires signature of someone who can legally bind your organization
- 4) Receive **Neuromorphic Research Cloud** accounts
You get a private VM on our system + accounts for your team members
- 5) Request **Loihi hardware**
We'll loan you physical systems when and if you need them...

Loihi Systems

Q4 2017

Wolf Mountain

Remote Access
4 Loihi/Board



Q2 2018

Nahuku

Arria10 Expansion Board
For cloud & local use
8-32 Loihi/Board



Q3 2018

Kapoho Bay

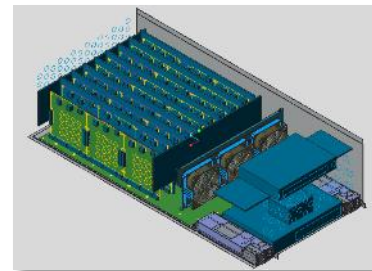
1-2 Loihi
DVS interface
USB host interface



Q2 2019

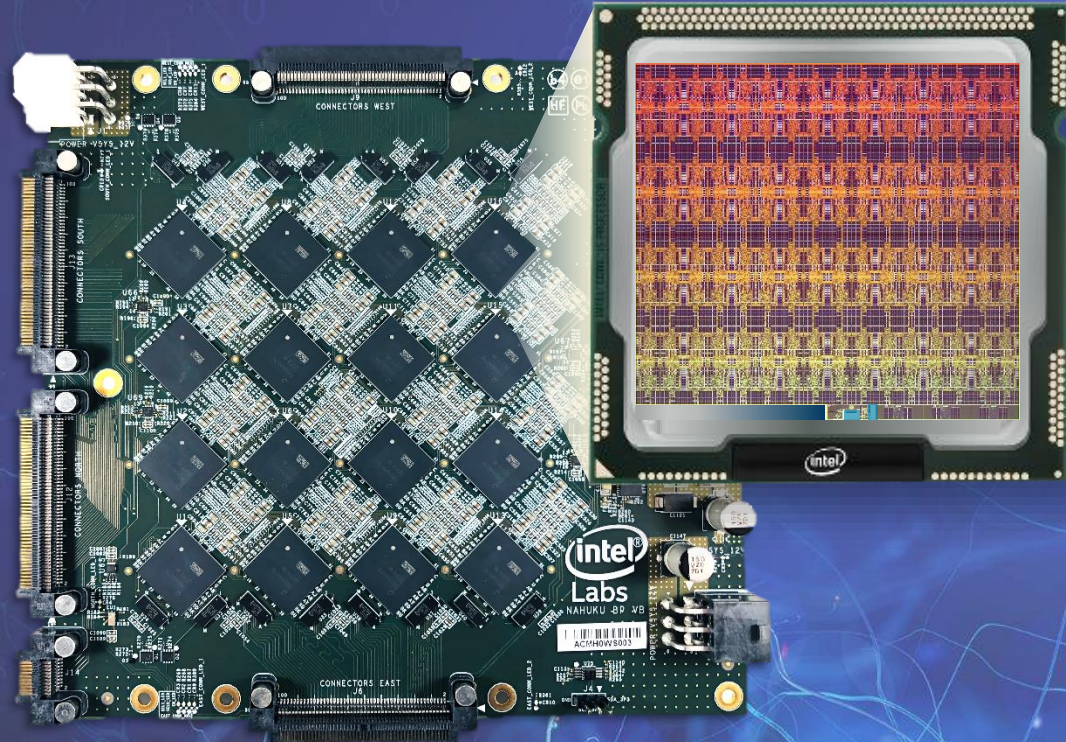
Pohoiki Springs

Remote Access
Up to 768 chips
(100M neurons)

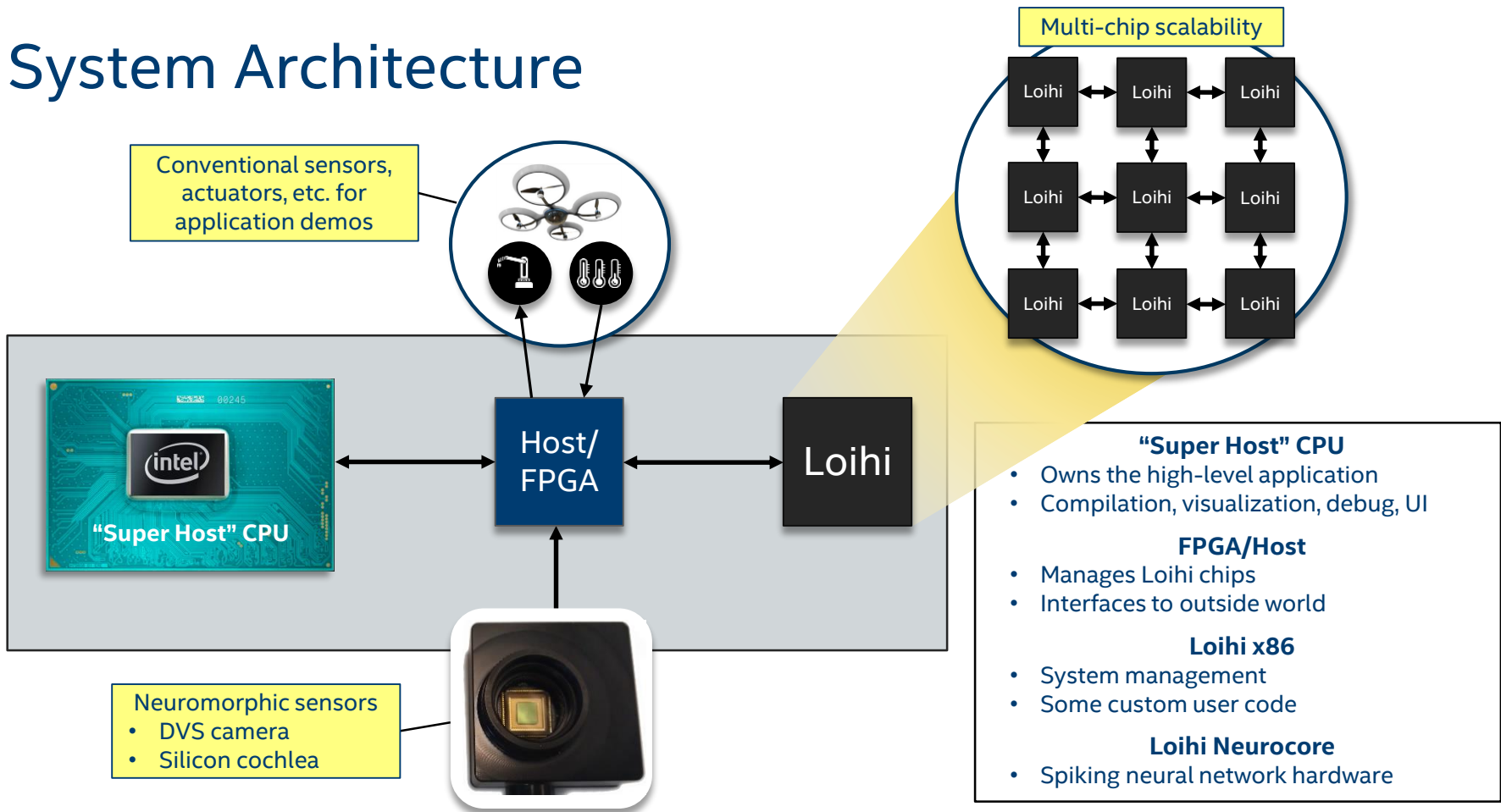


NAHUKU 32-CHIP PLATFORM

- 32 Loihi chips
- 4 x 4 mesh of chips
- Top and bottom sides

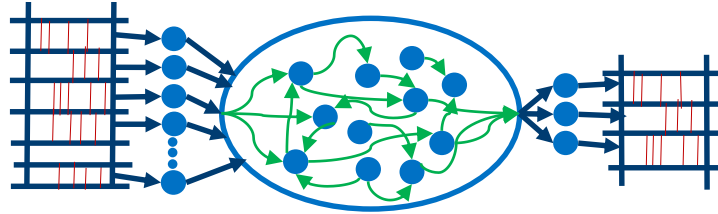


System Architecture



SDK

Architectural Considerations



Abstract Network
Definition

Objective

efficiently map abstract spiking neural network definitions onto our heterogeneous hierarchical implementation

Software
Development Kit

Architectural Principles

Programmability – Must be accessible in the lingua franca of machine learning (python) and at multiple levels of abstraction.

Simplicity and Modularity– Hardware details are abstracted away. Easy things are easy and complexity is added incrementally. Functionality available through modular interfaces.

Efficiency and Scalability – Additional hardware resource retains efficiency and adds to performance and scalability not to complexity.

Observability – Rich ability to probe and monitor networks as they execute.

Flexibility/Extensibility– Designed to snap into higher level APIs and enable a variety of sensors and actuators.



Loihi System

Heterogeneous,
Hierarchical Hardware

Nx SDK Software Architecture

A module is a complete NxNet defined algorithm (I/O, documentation, etc.)

Computational Modules

LCA	LSNN	EPL	VSA	TPAM
SLIC	CSS	Path Planning	DNF	Astro

3rd party APIs and Frameworks

Nengo	EONS	NRP
ROS	Tensorflow	PyNN

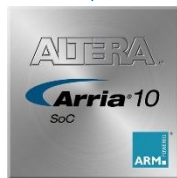
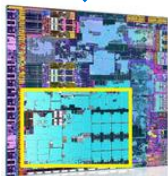
NxNet API

Sequential Neural Interfacing Processes (SNIPs)

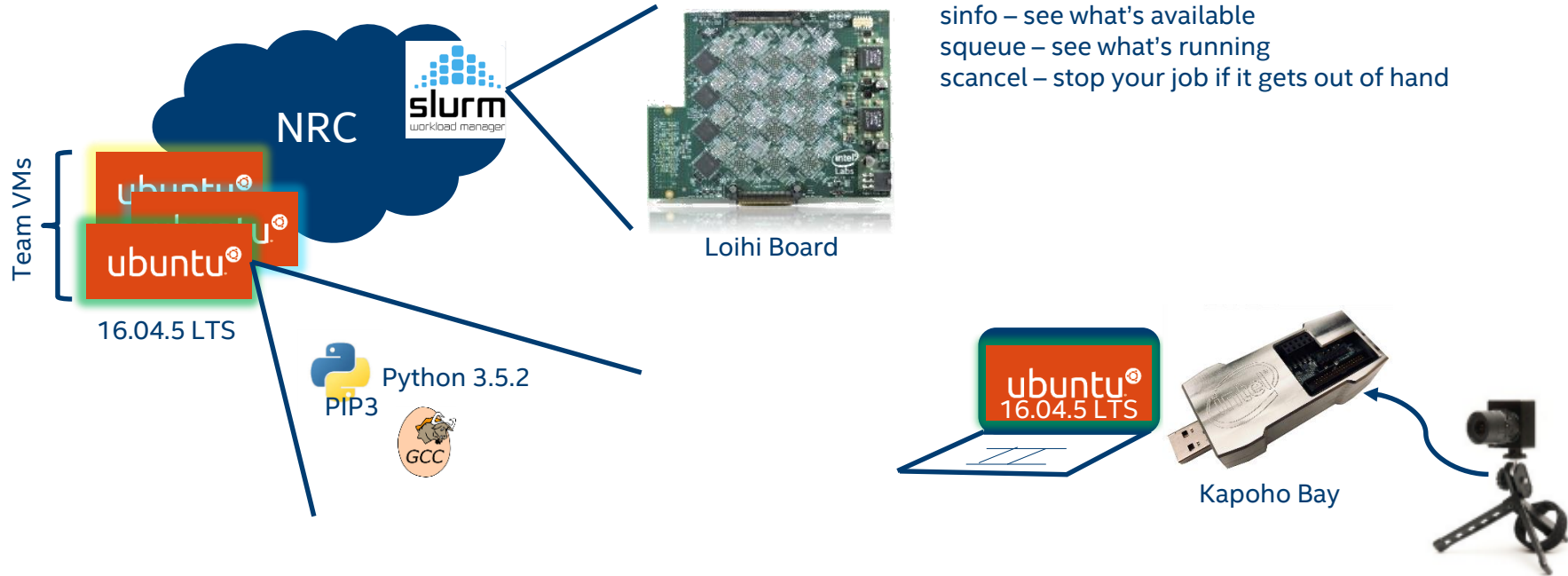
Spiking Neural Network (SNN)

NxCompiler

NxCore/NxDriver/NxRuntime

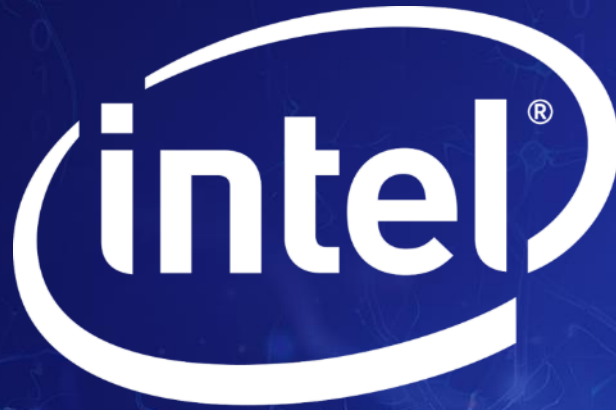


Toolchain



* Other names and brands may be claimed as the property of others

Thank You!



Email inrc_interest@intel.com for more information

LEGAL INFORMATION

This presentation contains the general insights and opinions of Intel Corporation ("Intel"). The information in this presentation is provided for information only and is not to be relied upon for any other purpose than educational. Intel makes no representations or warranties regarding the accuracy or completeness of the information in this presentation. Intel accepts no duty to update this presentation based on more current information. Intel is not liable for any damages, direct or indirect, consequential or otherwise, that may arise, directly or indirectly, from the use or misuse of the information in this presentation.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at [intel.com](https://www.intel.com), or from the OEM or retailer.

No computer system can be absolutely secure. No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document. Intel, the Intel logo, Movidius, Core, and Xeon are trademarks of Intel Corporation in the United States and other countries.

*Other names and brands may be claimed as the property of others

Copyright © 2019 Intel Corporation.