Welcome!

Thank you for joining us today! As we wait for everyone to get settled, we'd like to bring a few things to your attention:

1. This webinar is being recorded. The recording will be available via PSC's YouTube channel and the Neocortex website over the next few days.
2. There will be 40 minutes of presentation followed by Q&A. To maintain a quality experience for everyone, please mute your microphone during the presentations.
3. We hope you will participate in this interactive webinar by:
    - Asking and upvoting questions to our team via the Zoom Q&A. The *slido* link was closed at this time.
    - Completing the Zoom polls that will appear during the webinar.

    Your questions will seed the Q&A session in the final 20 minutes.
4. This webinar abides to the XSEDE code of conduct.

PSC

NSF

XSEDE Code of Conduct

XSEDE has an external code of conduct which represents our commitment to providing an inclusive and harassment-free environment in all interactions regardless of race, age, ethnicity, national origin, language, gender, gender identity, sexual orientation, disability, physical appearance, political views, military service, health status, or religion. The code of conduct extends to all XSEDE-sponsored events, services, and interactions.

Code of Conduct: https://www.xsede.org/codeofconduct

Contact:

- Event organizer: *PSC*
- XSEDE ombudspersons:
  - Linda Akli, Southeastern Universities Research Association (akli@sura.org)
  - Lizanne Destefano, Georgia Tech (lizanne.destefano@ceismc.gatech.edu)
  - Ken Hackworth, Pittsburgh Supercomputing Center (hackworth@psc.edu)
  - Bryan Snead, Texas Advanced Computing Center (jbsnead@tacc.utexas.edu)
- Anonymous reporting form available at https://www.xsede.org/codeofconduct.

Technical Overview of the Cerebras CS-1, the AI Compute Engine for Neocortex

August 19, 2020

NEOCORTEX

Unlocking Interactive AI for Rapidly Evolving Research

Natalia Vassilieva
Sr. Technical Product Manager
Cerebras Systems Inc.

PSC Team

Paola Buitrago
PI, Project Director & Executive Director

Nick Nystrom
Co-PI & AD for Scientific and Broader Impacts

Sergiu Sanielevici
Co-PI & AD for User Support

PSC

NSF Award 2005597

Outline

- About Neocortex
- Cerebras CS-1: Technical Overview
- Q&A

About Neocortex

Acquisition and operation of *Bridges*, *Bridges-AI*, *Bridges-2*, and ***Neocortex*** are made possible by the National Science Foundation:

NSF Award OAC-2005597 ($5M awarded to date):
Category II: Unlocking Interactive AI Development for Rapidly Evolving Research

Cerebras and HPE are delivering *Neocortex*

Neocortex – Project Goals

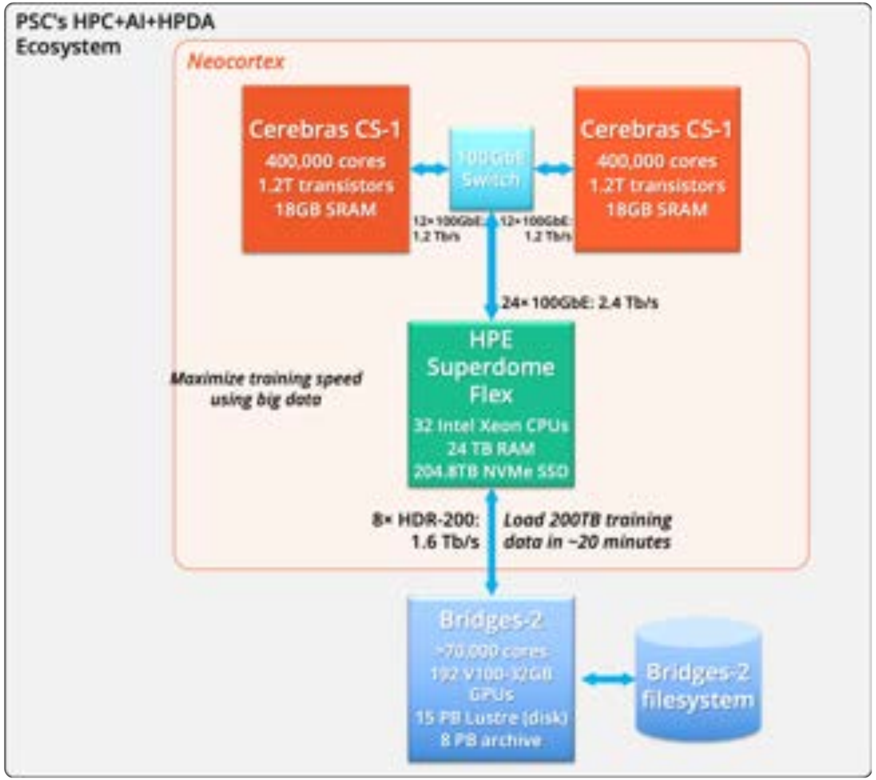
***Neocortex*, Unlocking Interactive AI Development for Rapidly Evolving Research**

A upcoming NSF funded advanced computing project with the following goals:

- Deploy *Neocortex* in 2020 and offer the national open science community revolutionary hardware technology to accelerate AI training at unprecedented levels.
- Research, explore, support and operate *Neocortex* for 5 years.
- Engage a wide audience and foster adoption of innovative technologies.

Neocortex: System Overview

NEOCORTEX
Unlocking Interactive AI for Rapidly Evolving Research



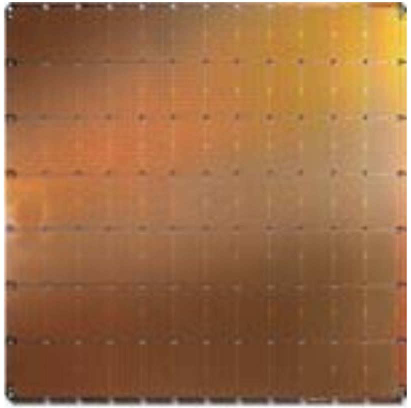
Cerebras CS-1: the AI Compute Engine for Neocortex

Technical Overview

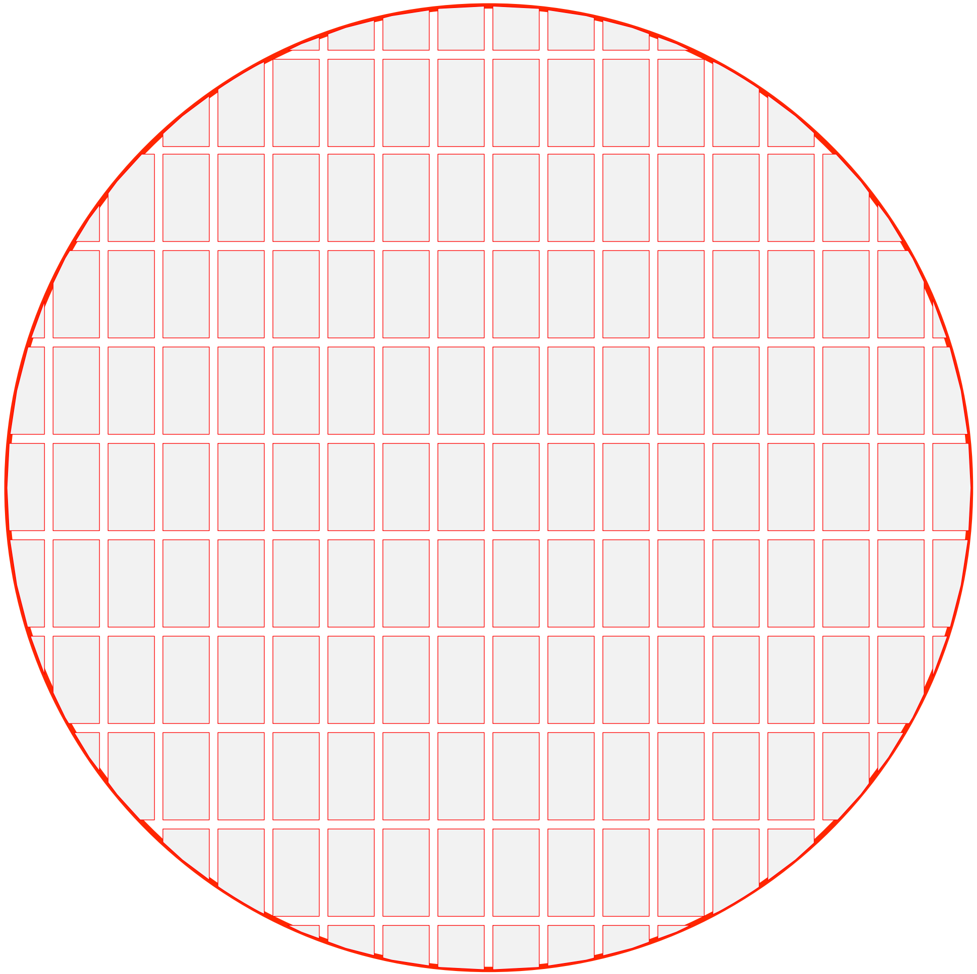
The CS-1 Solution

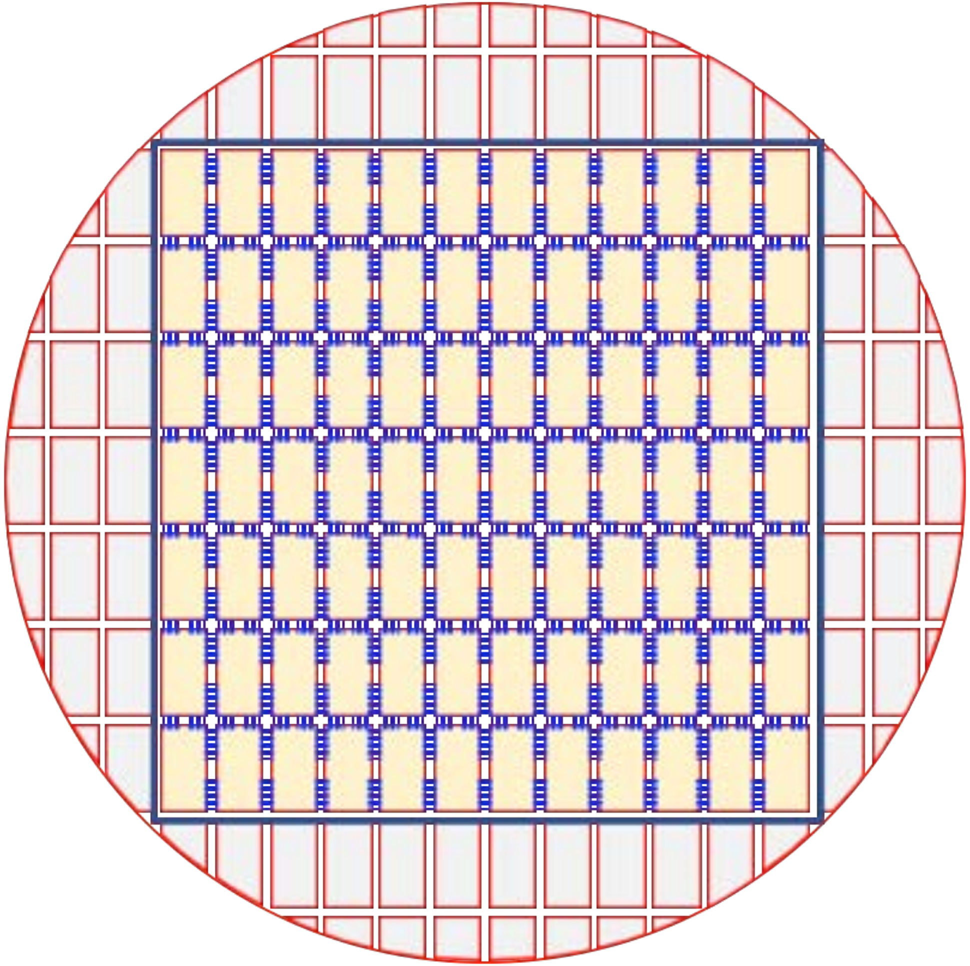
CS-1 System

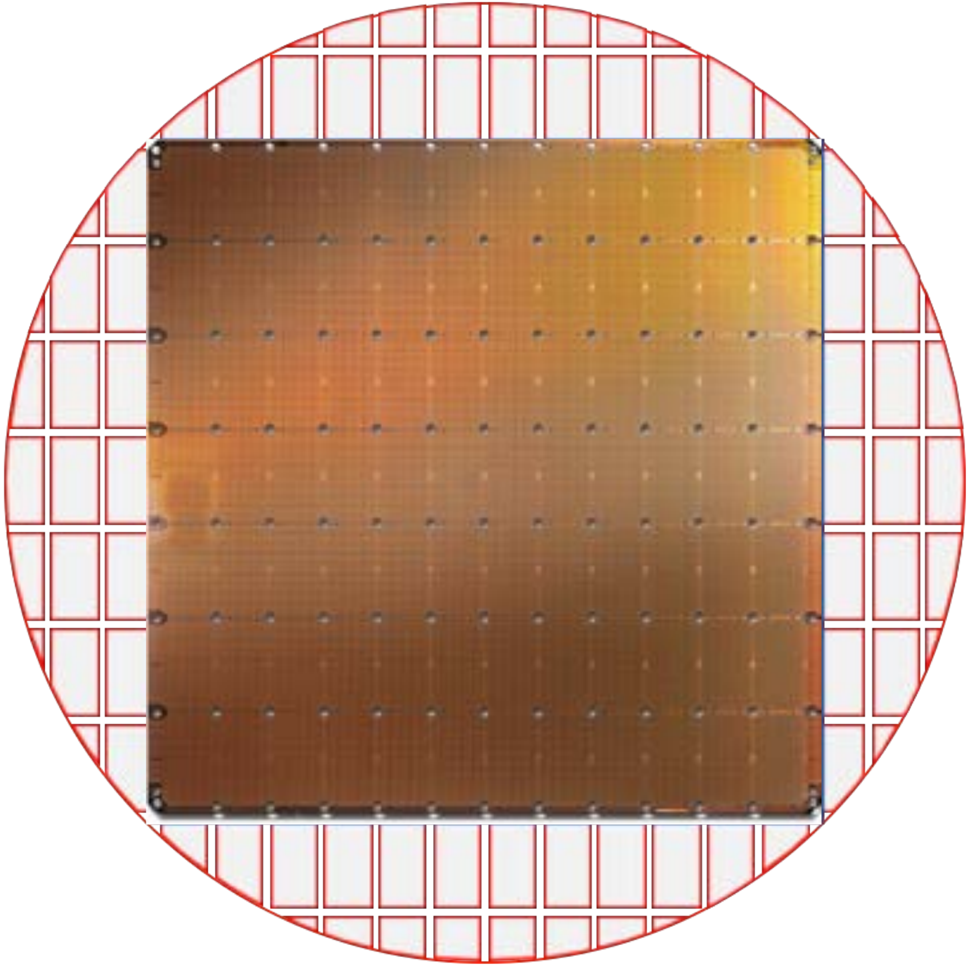
Wafer Scale Engine



Cerebras Software Platform







Cerebras Wafer Scale Engine (WSE)

The Most Powerful Processor for AI

400,000 AI-optimized cores

46,225 mm^2 silicon

1.2 trillion transistors

18 Gigabytes of On-chip Memory

9 PByte/s memory bandwidth

100 Pbit/s fabric bandwidth

TSMC 16nm process

Cerebras CS-1: Cluster-Scale DL Performance in a Single System

System processor: 1 x WSE

System IO: 12 x 100 GbE

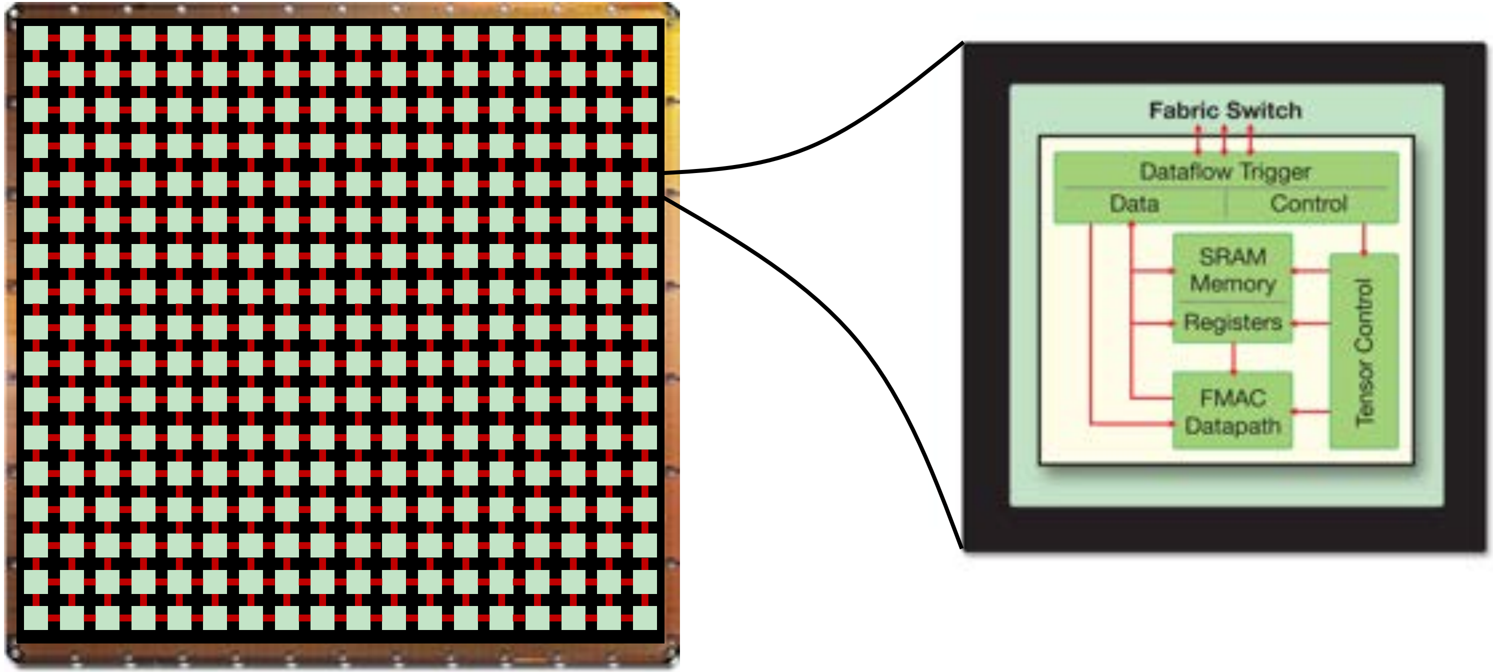
System power: 20 kW

Programming: using TensorFlow, PyTorch, and other frameworks

Built from the ground up for AI acceleration

The Wafer-Scale Engine (WSE)

2D Mesh of 400,000 Fully Programmable Processing Elements



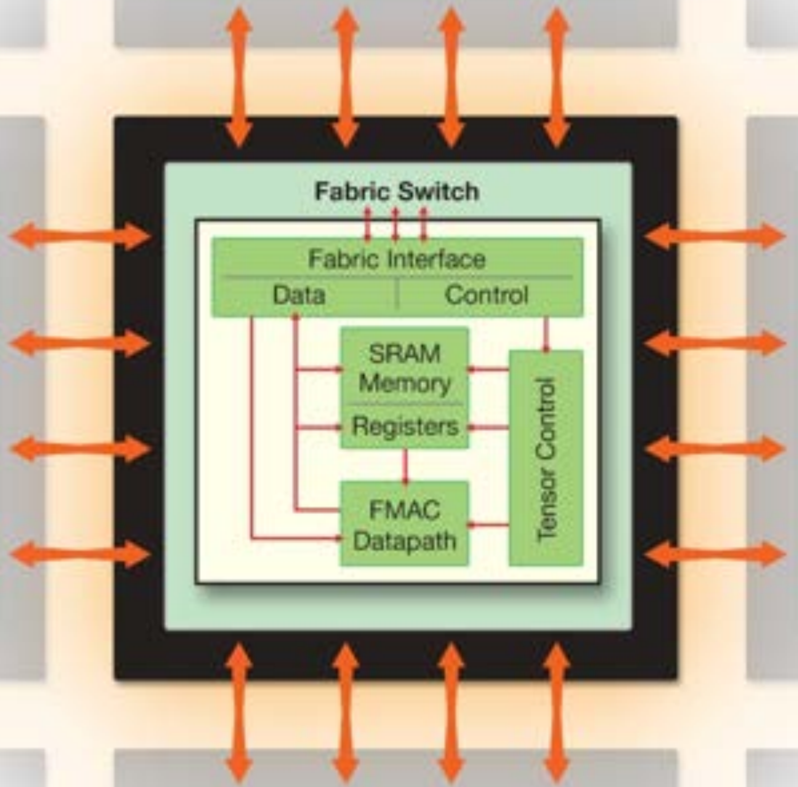
Designed for Deep Learning

Each component optimized for Deep Learning

Compute
- Fully-programmable core, ML-optimized extensions
- Dataflow architecture for sparse, dynamic workloads

Memory
- Distributed, high performance, on-chip memory

Communication
- High bandwidth, low latency fabric
- Cluster-scale networking on chip
- Fully-configurable to user-specified topology



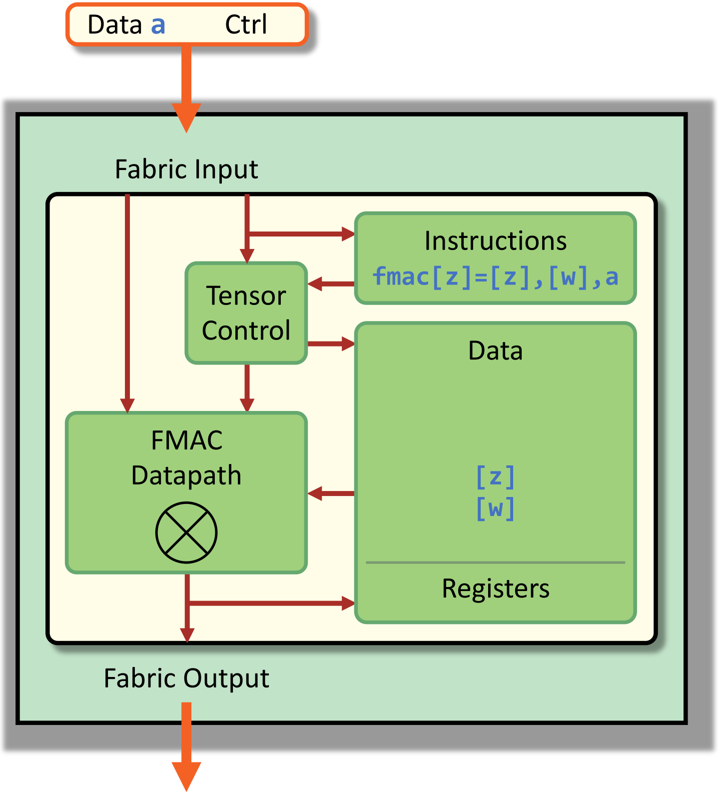
Designed for Sparsity

Dataflow scheduling in hardware

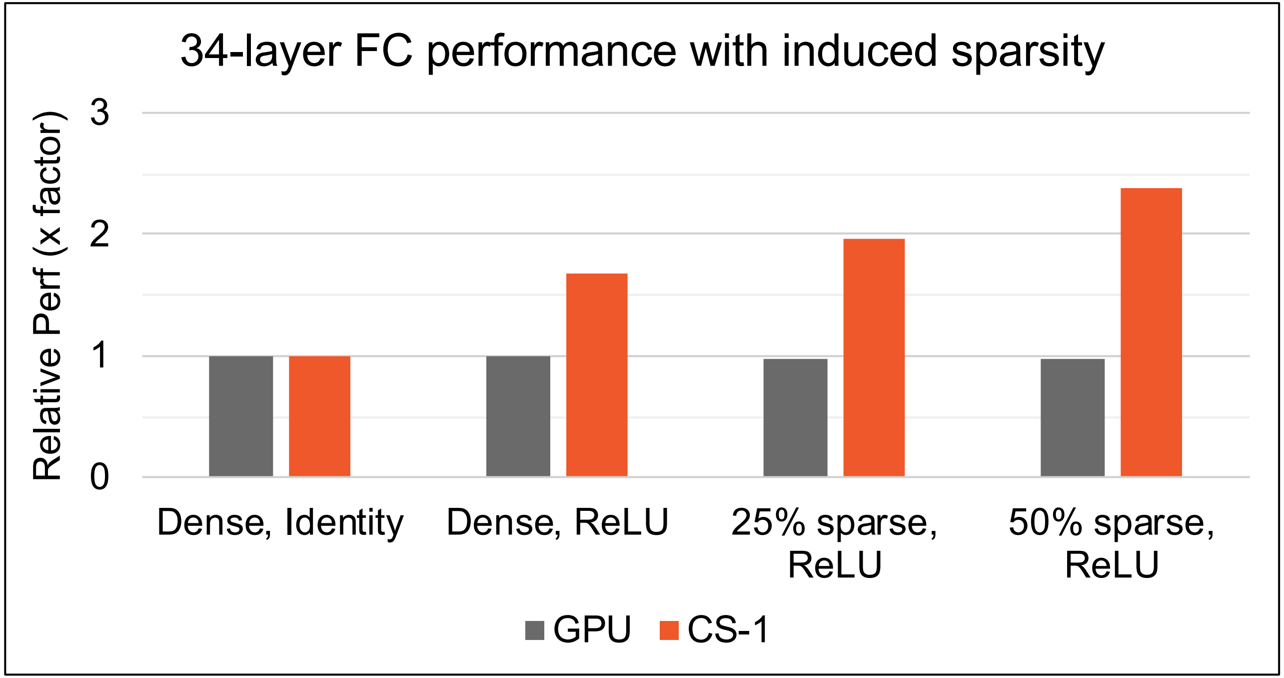
- Data and control received from fabric
- Triggers instruction lookup
- State machine schedules datapath cycles
- Output is written back to memory or fabric

Intrinsic sparsity harvesting

- Sender filters out sparse zero data
- Receiver skips unnecessary processing



Sparsity = Speed-up



- **1.7x perf gain** with ReLU
- **2.4x perf gain** with ReLU+50% sparsity

Advantages for Deep Learning

Compute is …

- **Massive**, more than can fit on a traditional single die
- Optimized for **linear ops on sparse tensors**, to execute most common ops fast, to exploit sparsity in models and data
- **Flexible**, to support evolving models

Advantages for Deep Learning

Compute is …

- **Massive**, more than can fit on a traditional single die
- Optimized for **linear ops on sparse tensors**, to execute most common ops fast, to exploit sparsity in models and data
- **Flexible**, to support evolving models

Memory is …

- **Large, high-bandwidth, tightly coupled with compute**, so utilization doesn't depend on batch size

cerebras

Advantages for Deep Learning

Compute is …

- **Massive**, more than can fit on a traditional single die
- Optimized for **linear ops on sparse tensors**, to execute most common ops fast, to exploit sparsity in models and data
- **Flexible**, to support evolving models

Memory is …

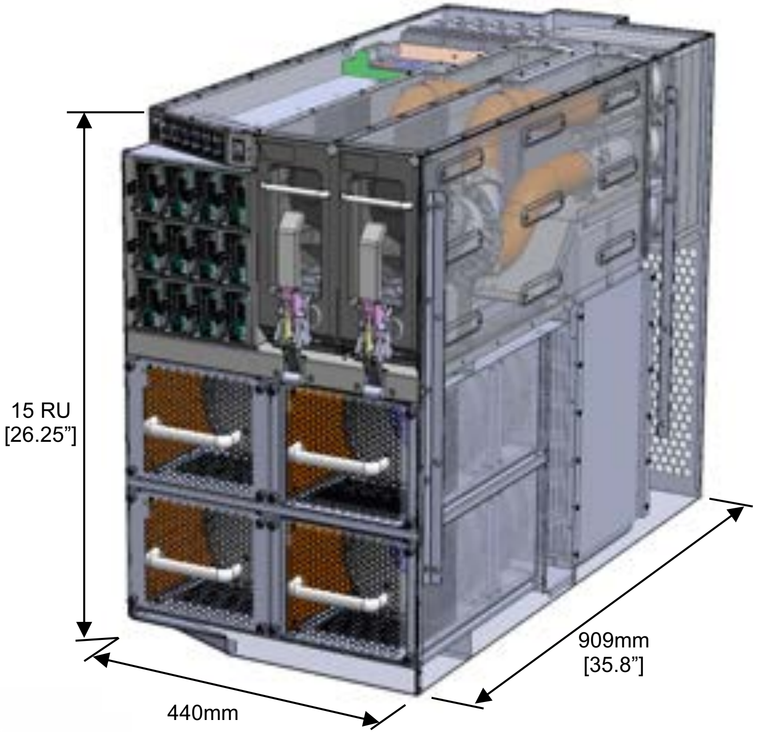
- **Large, high-bandwidth, tightly coupled with compute**, so utilization doesn't depend on batch size

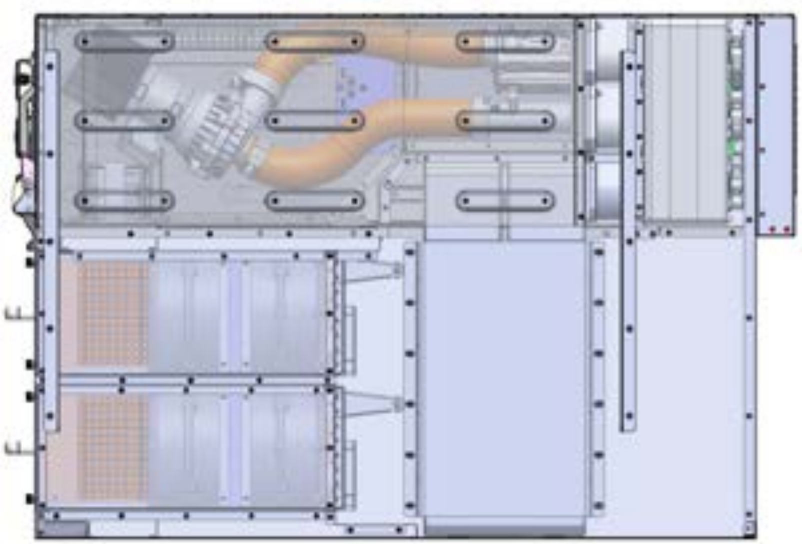
Fabric is …

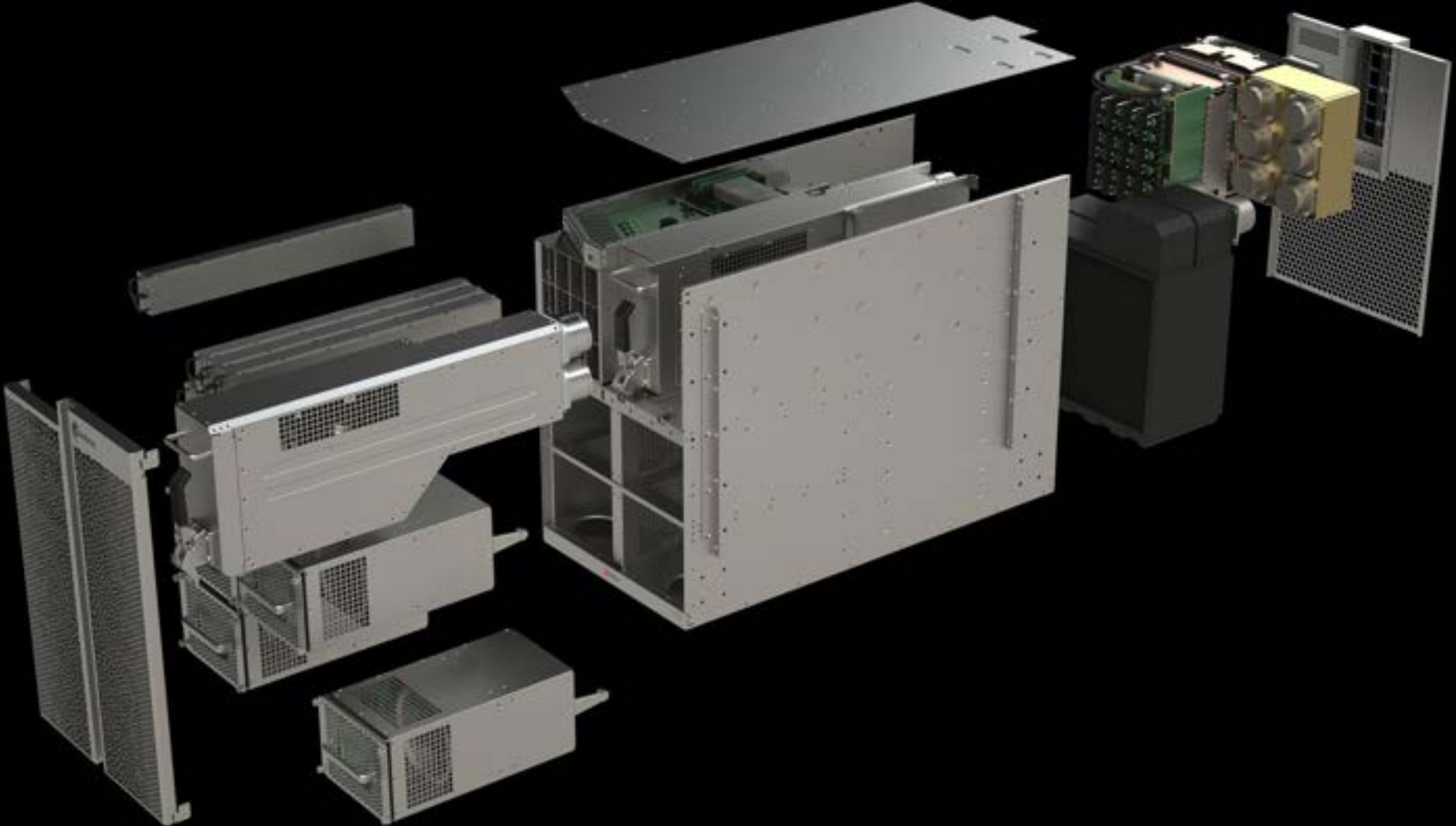
- **High bandwidth, low-latency** for seamless model and data parallelism
- **Fully configurable** for each workload

The CS-1

CS-1 System View

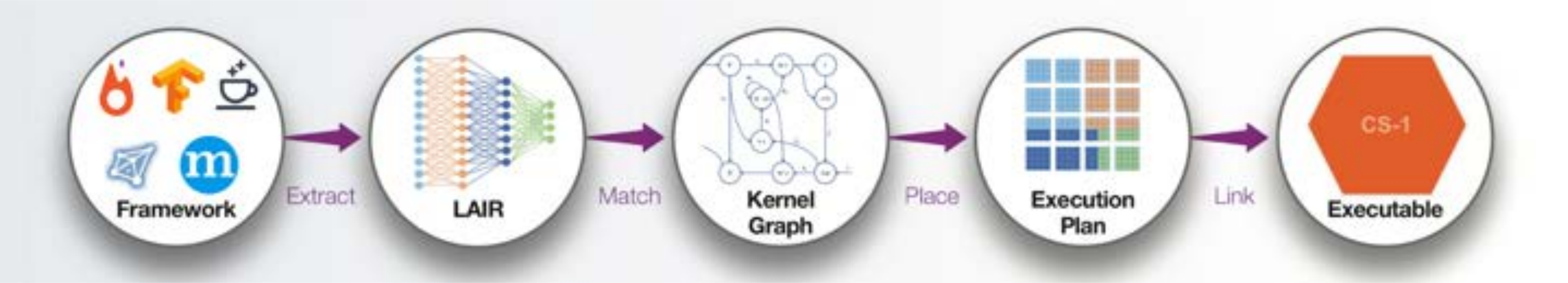






Software and Programming

Cerebras Software Stack handles graph compilation



- **Extract** graph representation of model from framework, convert to Cerebras IR
- **Match** computational subgraphs to kernels that implement portions of model
- **Place & Route** allocates compute and memory, assigns kernels to fabric sections, configures on-chip network
- **Link** creates executable output that can be loaded and run by CS-1

cerebras

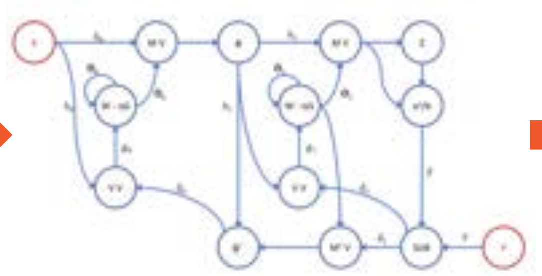
Extract Model from ML Framework, Convert to LAIR

Users program the WSE using standard ML frameworks, e.g. TensorFlow, PyTorch

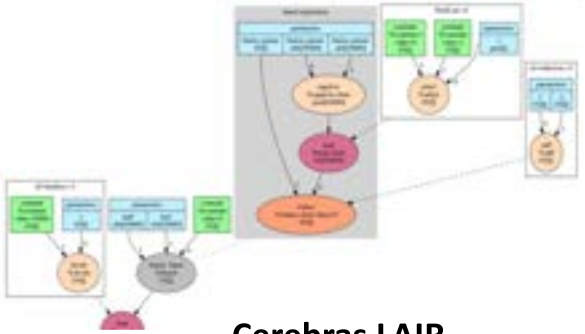
We extract graph representation of the model from the ML Framework and translate it into Cerebras LAIR (**L**inear **A**lgebra **I**ntermediate **R**epresentation).

TensorFlow

PyTorch



Framework IR



Cerebras LAIR

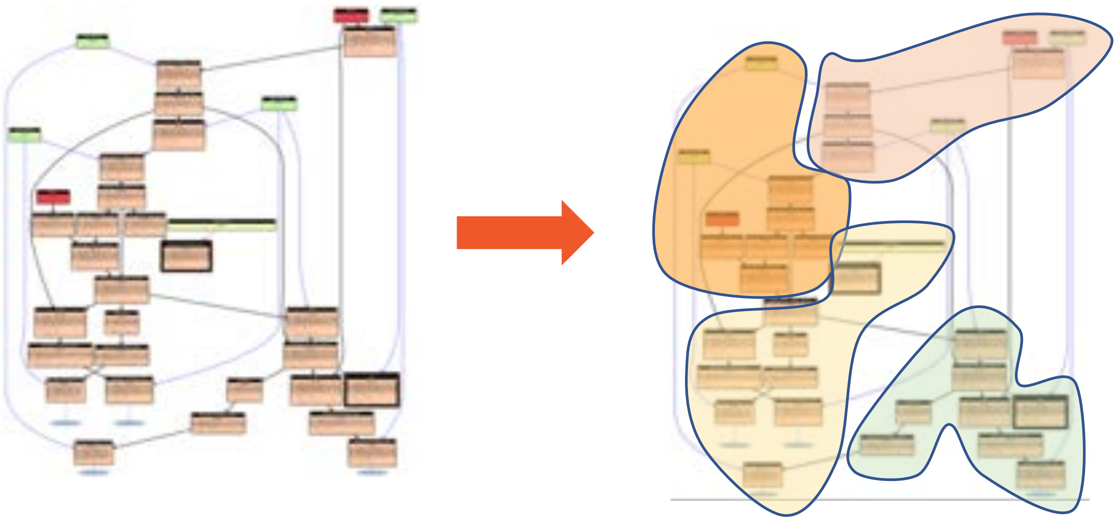
Match LAIR subgraphs to existing kernels

Subsections of the LAIR graph are **matched** to optimized microcode **kernels** in our high-performance kernel library.

Operational Graph

Match LAIR subgraphs to existing kernels

Subsections of the LAIR graph are **matched** to optimized microcode **kernels** in our high-performance kernel library.

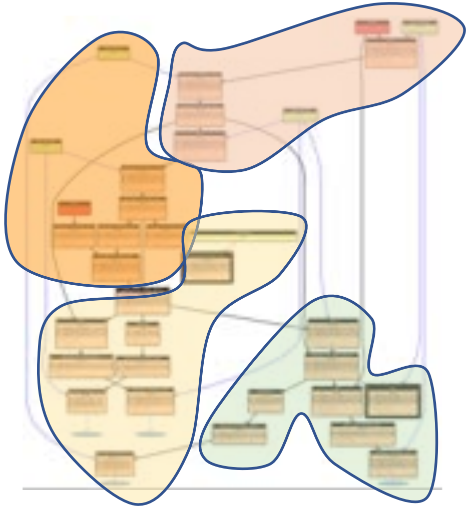


Operational Graph

Match LAIR subgraphs to existing kernels

Subsections of the LAIR graph are **matched** to optimized microcode **kernels** in our high-performance kernel library.

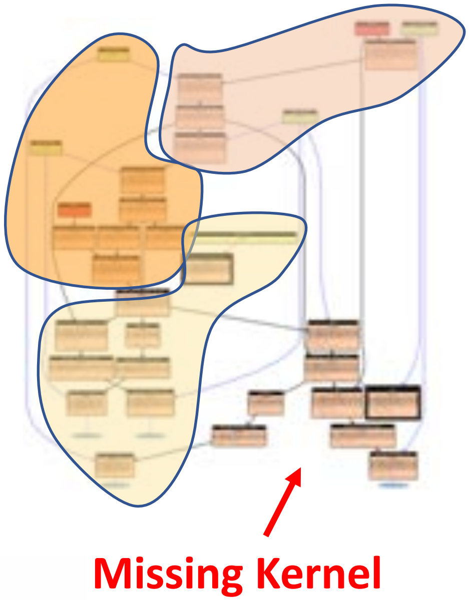
Operational Graph



Kernel Graph

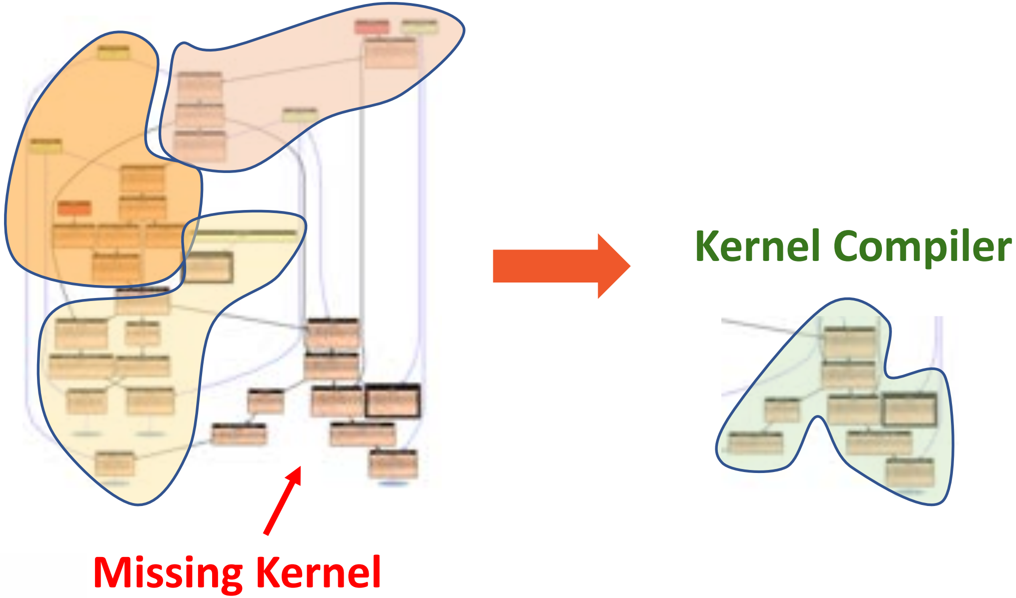
Match: Generate Missing Kernels

If no matching kernel exists in the optimized kernel library, the Cerebras **Kernel Compiler generates one dynamically** from the IR



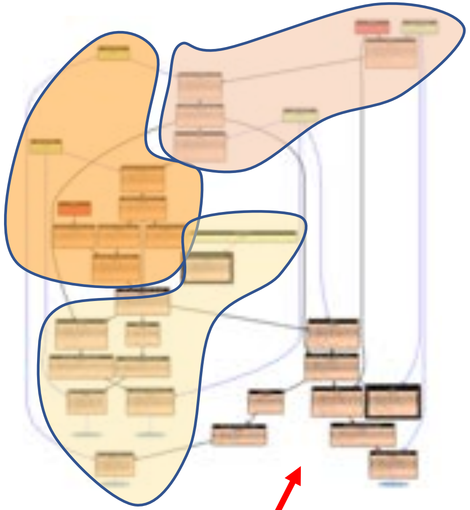
Match: Generate Missing Kernels

If no matching kernel exists in the optimized kernel library, the Cerebras **Kernel Compiler generates one dynamically** from the IR



Match: Generate Missing Kernels

If no matching kernel exists in the optimized kernel library, the Cerebras **Kernel Compiler generates one dynamically** from the IR



Missing Kernel

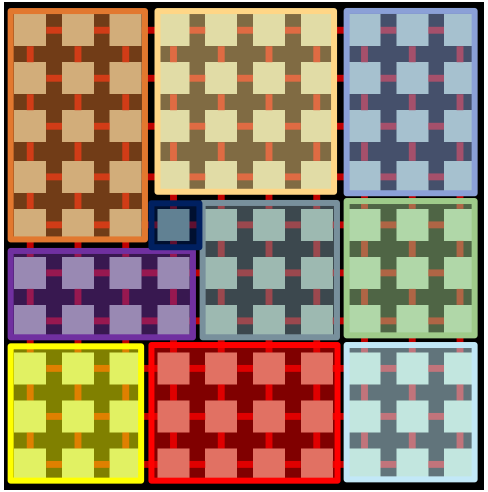
Kernel Compiler

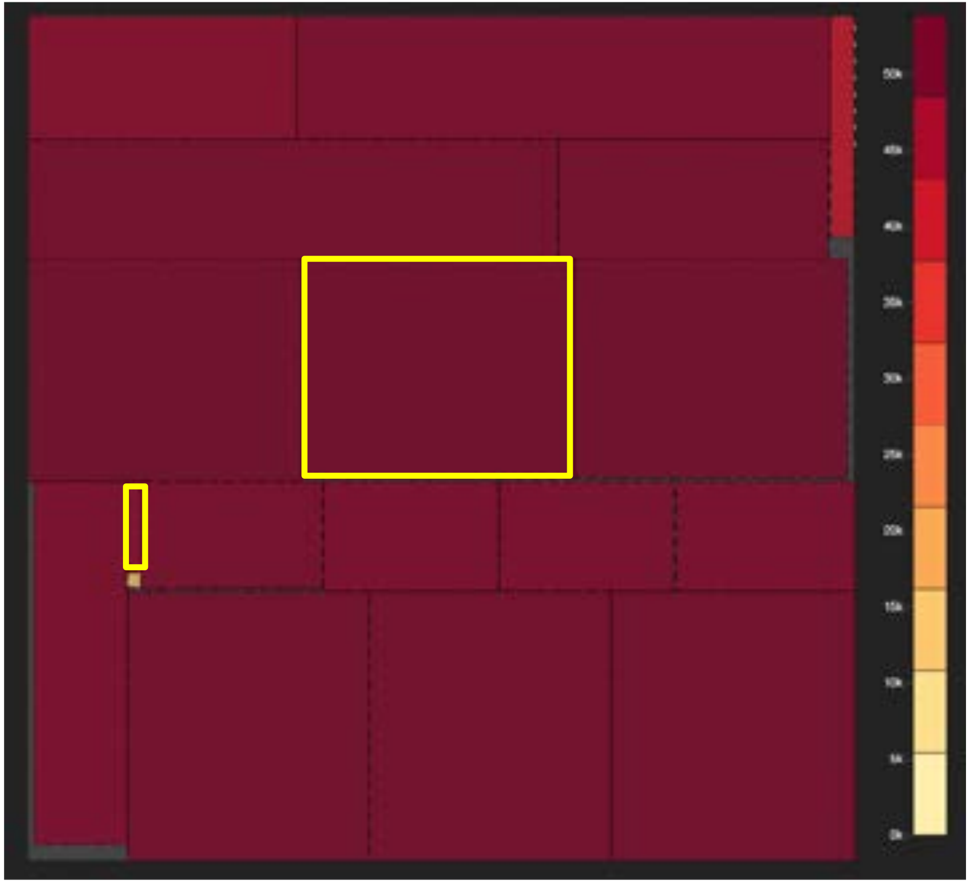
```
// initialize the global registers
mov16 r_recv_flag = [recv_flag]
mov16 r_trans_flag = [trans_flag]
mov16 r_recv_counter_local_k = [recv_counter_local_k]
mov16 r_trans_counter_local_k = [trans_counter_local_k]
mov16 r_curr_recv_range_begin = [curr_recv_range_begin_x]
mov16 r_curr_trans_range_begin = [curr_trans_range_begin_x]

.auto_gpr r_reduce_x_phase
mov16 r_reduce_x_phase = [reduce_x_phase]
.auto_gpr r_base
.auto_gpr r_weight
```

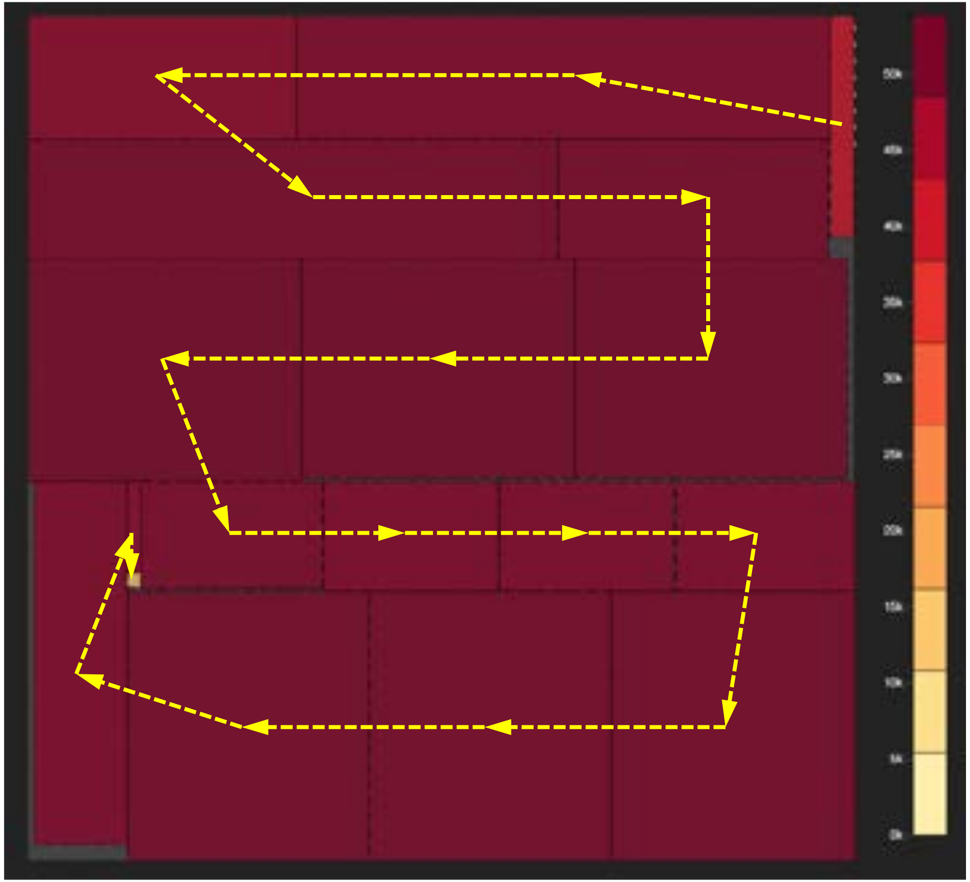
Synthesized Kernel

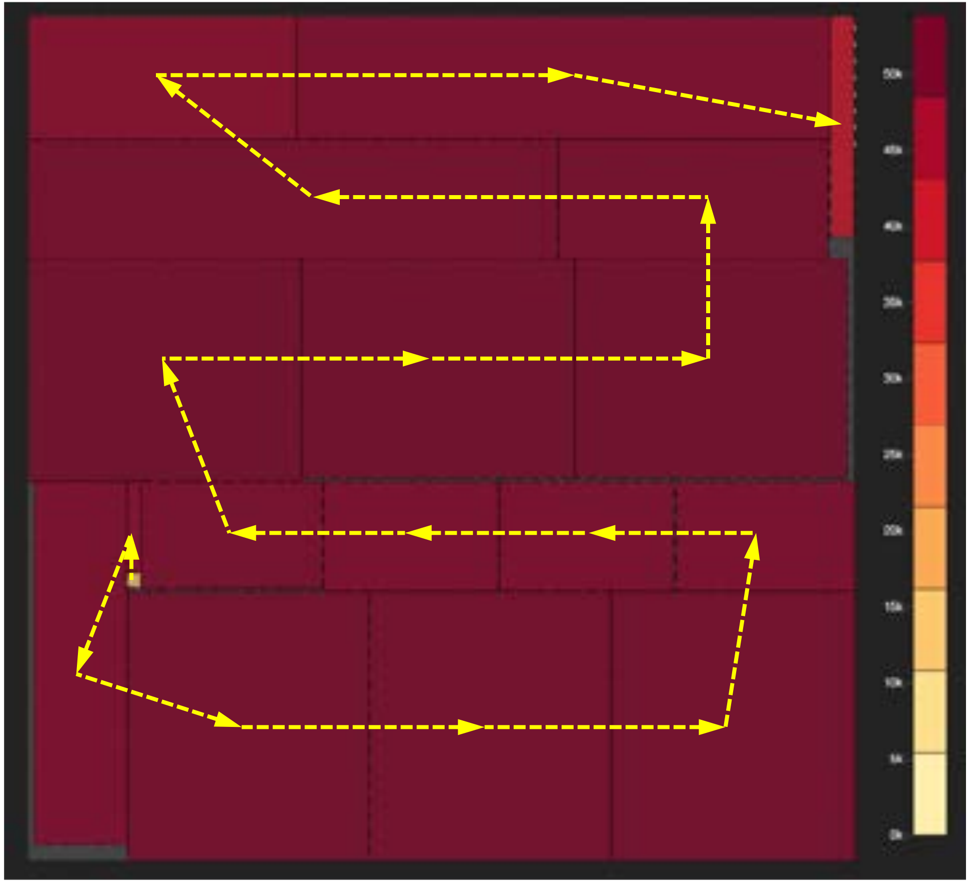
Place Kernels & **Route** On-Chip Network





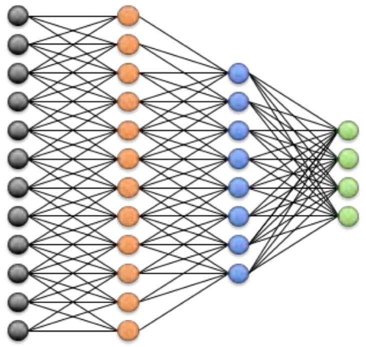
50k
45k
40k
35k
30k
25k
20k
15k
10k
5k
0k

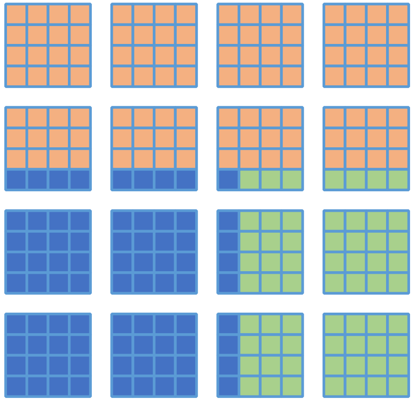

50k
45k
40k
35k
30k
25k
20k
15k
10k
5k
0k


50k
45k
40k
35k
30k
25k
20k
15k
10k
5k
0k

Summary: Compiler & Software stack

1. Graph is **extracted** from ML Framework into LAIR
2. Linear algebra **kernels are matched** to subsections of the graph
3. Kernels are **sized and placed** on the chip to **balance throughput** of all layers
4. Network fabric is **routed** to provide shortest-path, full bandwidth communication
5. Key kernels are **hand-optimized** in assembly
6. Other kernels can be **written** using our Kernel API





cerebras

CS-1 is designed to unlock smarter techniques and scale

CS-1 has a data flow architecture

- Flexibility to stream *token by token*
- Inherent sparsity harvesting

CS-1 is a MIMD architecture

- Can program each core independently
- Perform different operations on different data

CS-1 was built to **enable the next generation of models** otherwise limited today.

CS-1 advantage vectors

- Massive, accessible **performance on a single system**
  - System Advantage: Avoids communication bottlenecks.
  - System Advantage: Model-parallel training scales seamlessly
  - Usability Advantage: No orchestration/sync headaches
  - ML Advantage: Train with small batches at high utilization
  - ML Advantage: Avoid tricky learning rate schedules and optimizers
- **Flexibility** for new models and training methods
  - Uniquely advantaged for novel smart techniques, e.g. sparsity, conditional computations
- **Ultimate performance** with a cluster of CS-1
  - Easier to scale to fewer fatter nodes
  - High-bandwidth interconnect between nodes

CS-1 in Summary

Built from the ground up to **accelerate deep learning** by orders of magnitude and **empower researchers and ML practitioners** to do more, faster.

To Learn More and Participate

NEOCORTEX
Unlocking Interactive AI for Rapidly Evolving Research

Join our coming webinars	https://www.cmu.edu/psc/aibd/neocortex/event-list.html
Join the Early User Program (more info coming)	https://www.cmu.edu/psc/aibd/neocortex/early-user-program.html
Watch the Neocortex website for updates!	https://www.cmu.edu/psc/aibd/neocortex/
Contact us with additional questions, input, or requests	neocortex@psc.edu

Thank you to all those contributing to Neocortex!

NSF

cerebras

Hewlett Packard Enterprise

Andrew K. Adams
Paola Buitrago
Ken Hackworth
Ed Hanna
Dave Moses
Nick Nystrom
Rajanie Prabha
Sergiu Sanielevici
Amanda Slimick
Julian Uran
John Urbanic
Bryan Webb

NEOCORTEX

Unlocking Interactive AI for Rapidly Evolving Research