



Challenges in AI Chip Design

Olivier Temam

Director DC & Chip Research, Google DeepMind (GDM)

temam@google.com

MASSIVE VELOCITY CHALLENGES

Overview

AI/HW VELOCITY GAP

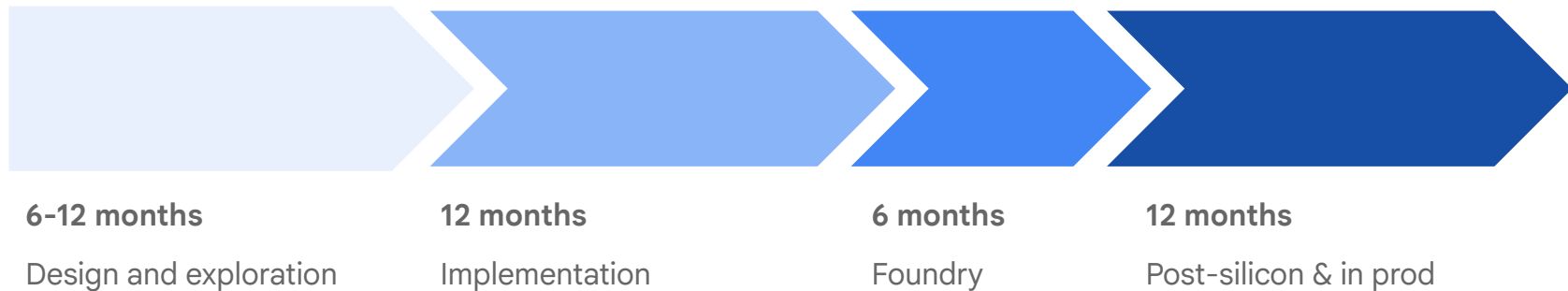
NO TIME TO UNDERSTAND NEW AI MODELS IN DEPTH

NO TIME TO VALIDATE NEW HARDWARE IDEAS

AI / HARDWARE VELOCITY GAP

Massive **velocity gap**: AI algorithms evolution / HW design

HW



Massive **velocity gap**: AI algorithms evolution / HW design

HW



6-12 months

Design and exploration

12 months

Implementation

6 months

Foundry

12 months

Post-silicon & in prod

AI



3-6 months ($\pm$ 3 months !)

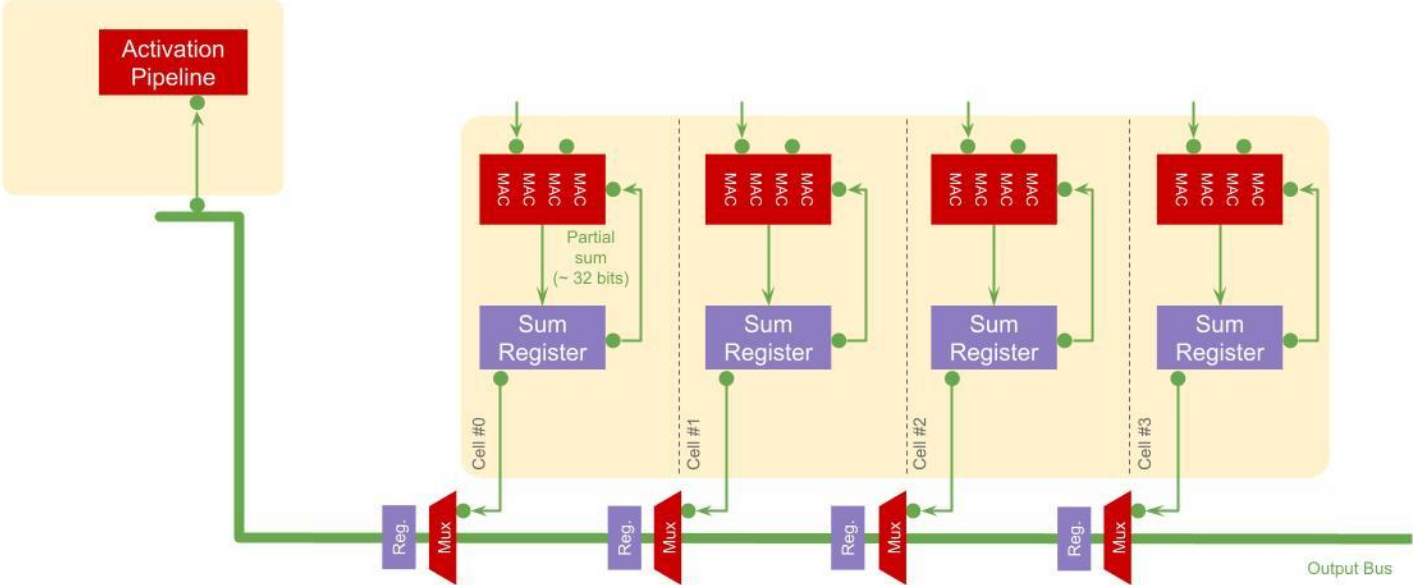
Significant algorithm
evolution with potential
HW impact

Massive **velocity gap**: AI algorithms evolution / HW design

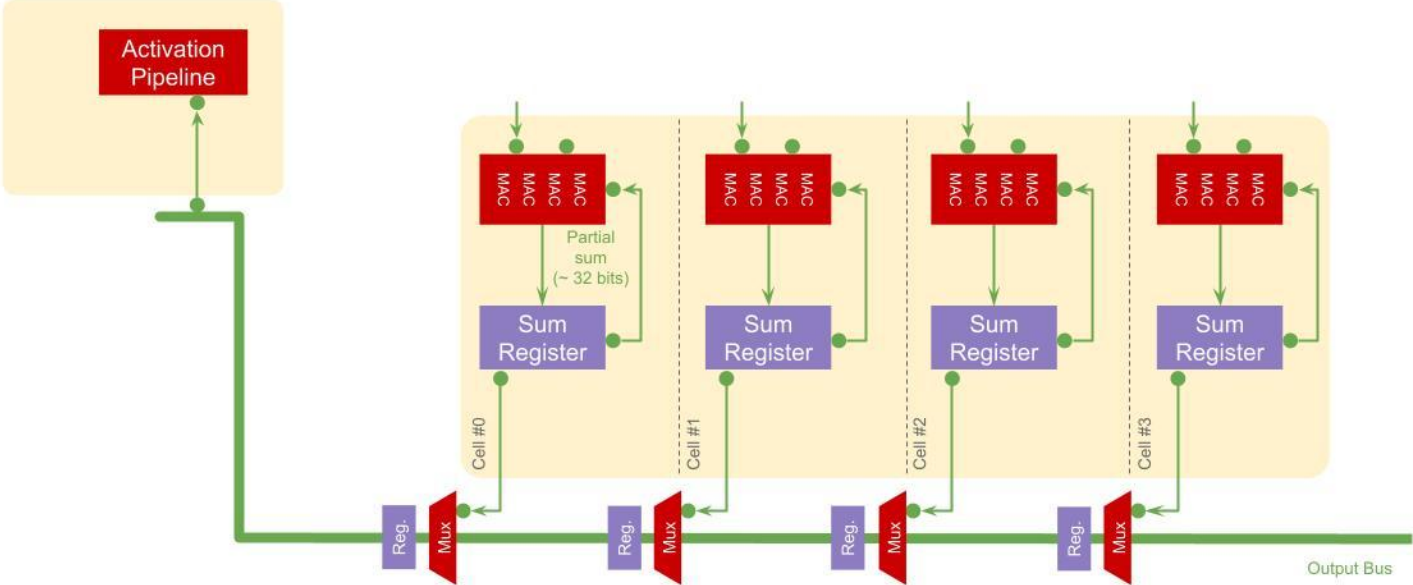
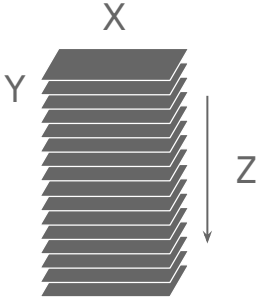
Often feels like...



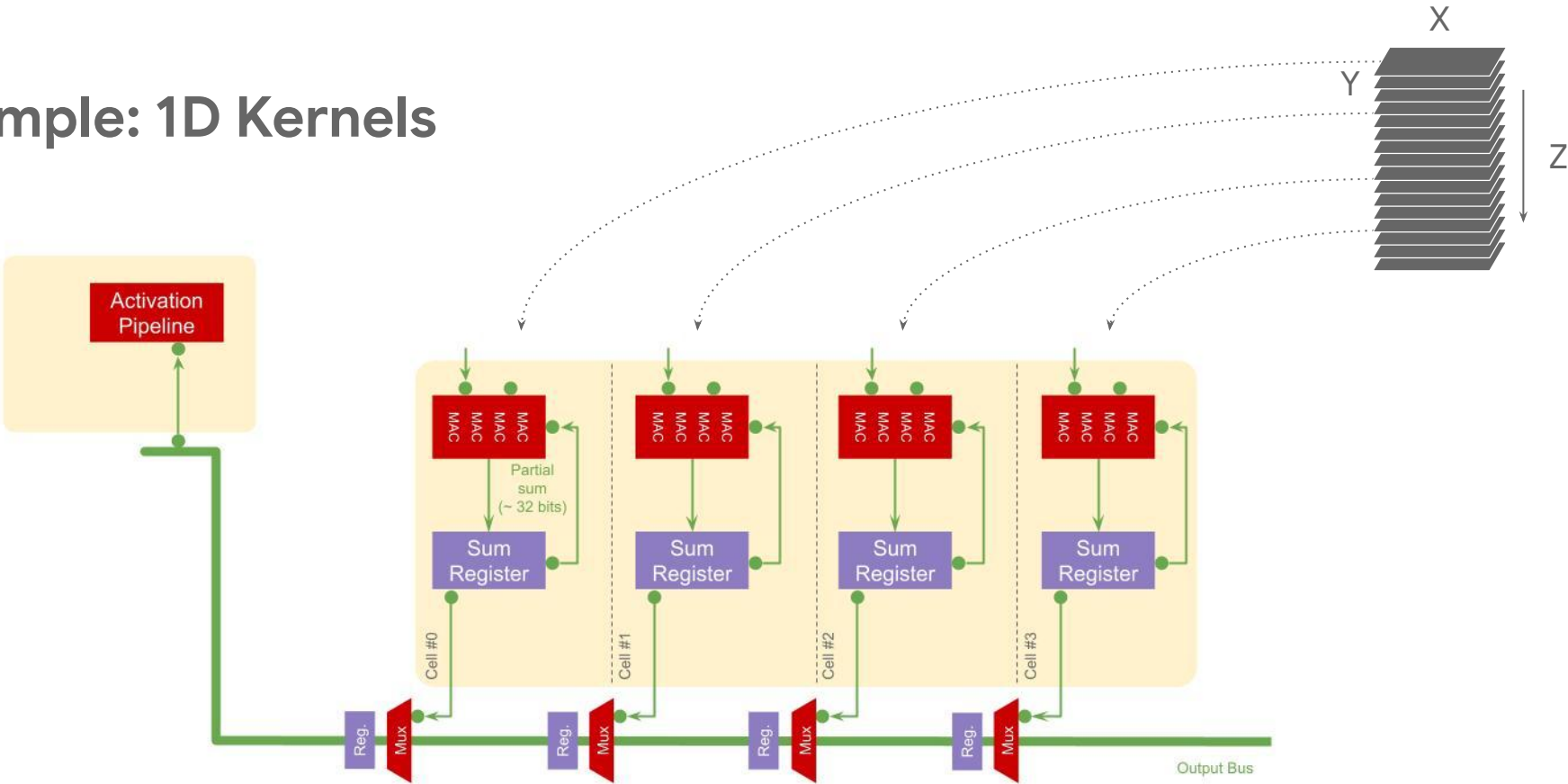
Example: 1D Kernels



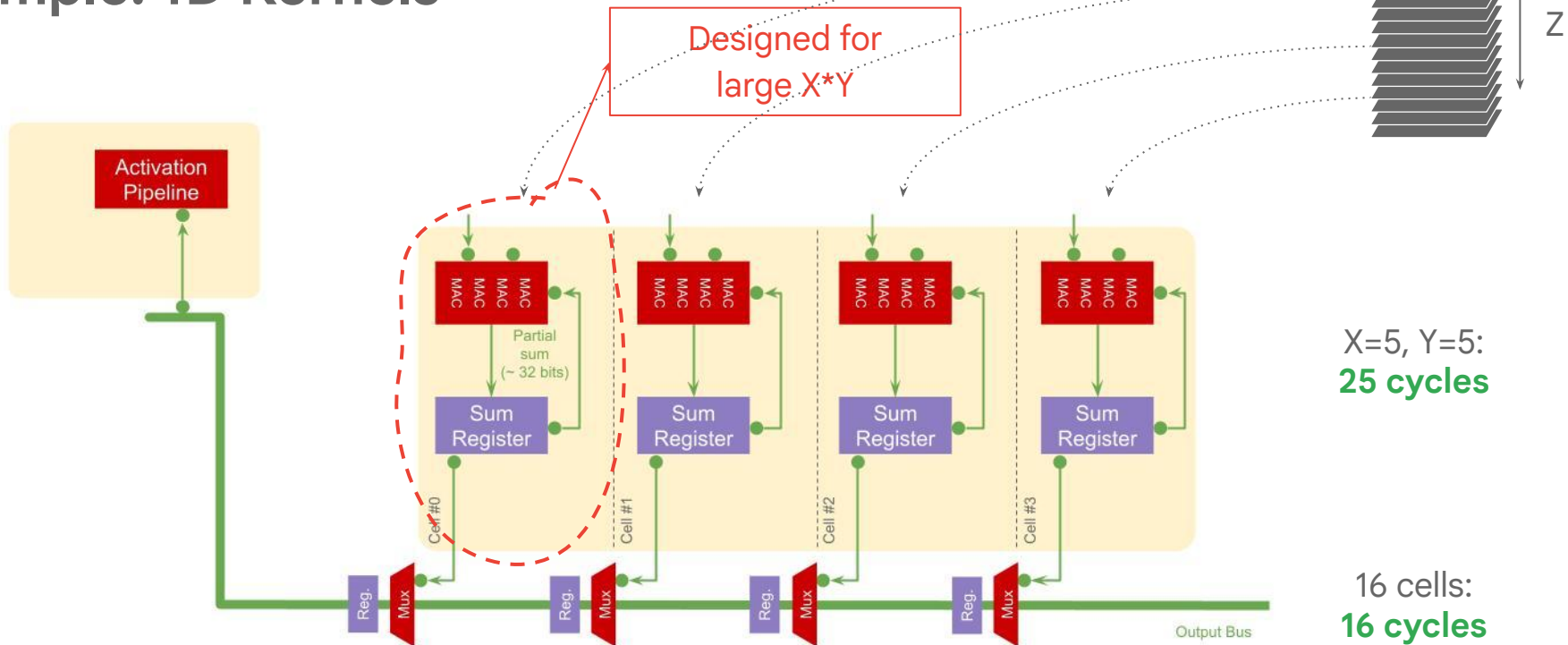
Example: 1D Kernels



Example: 1D Kernels



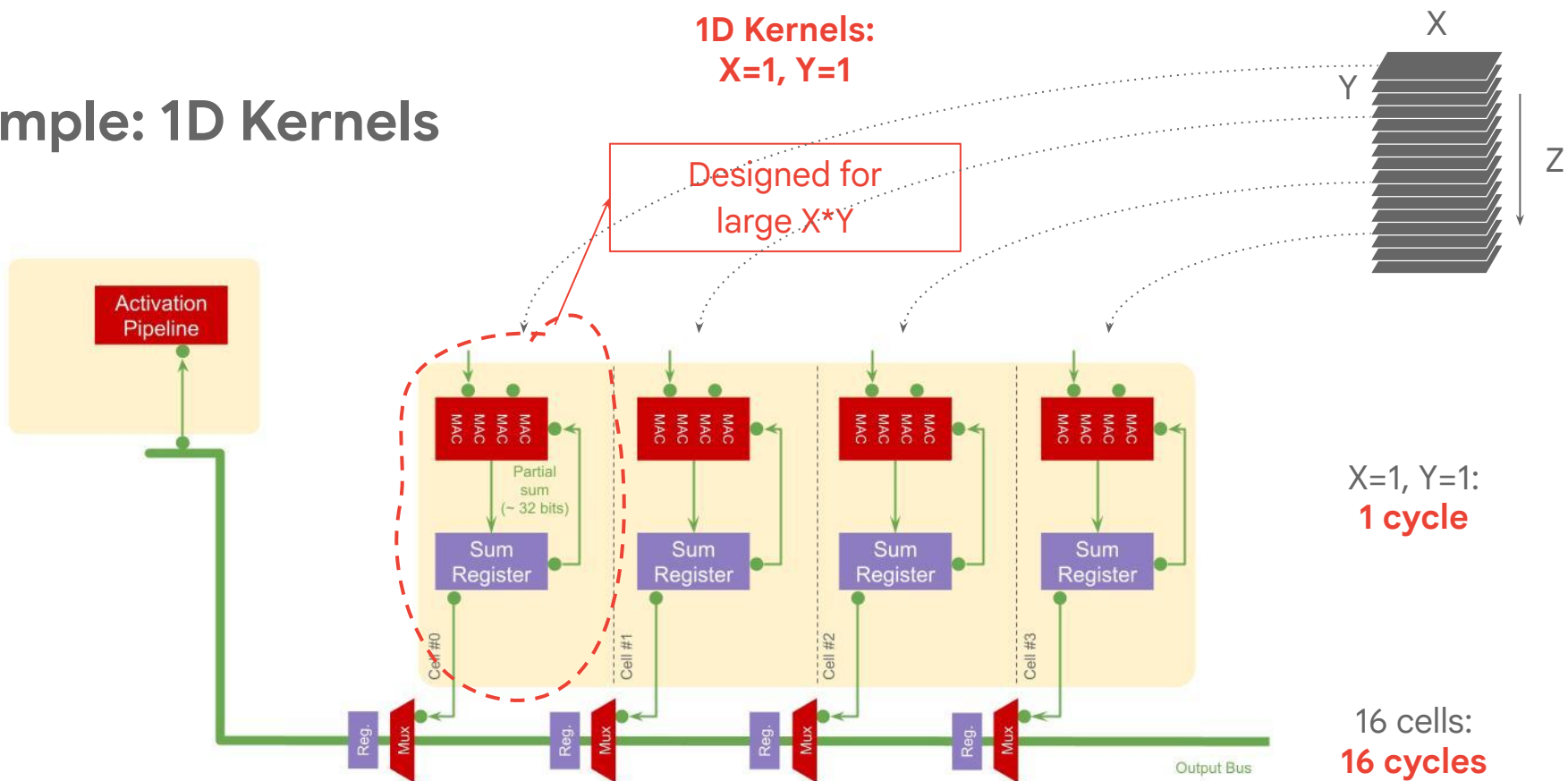
Example: 1D Kernels



Example: 1D Kernels

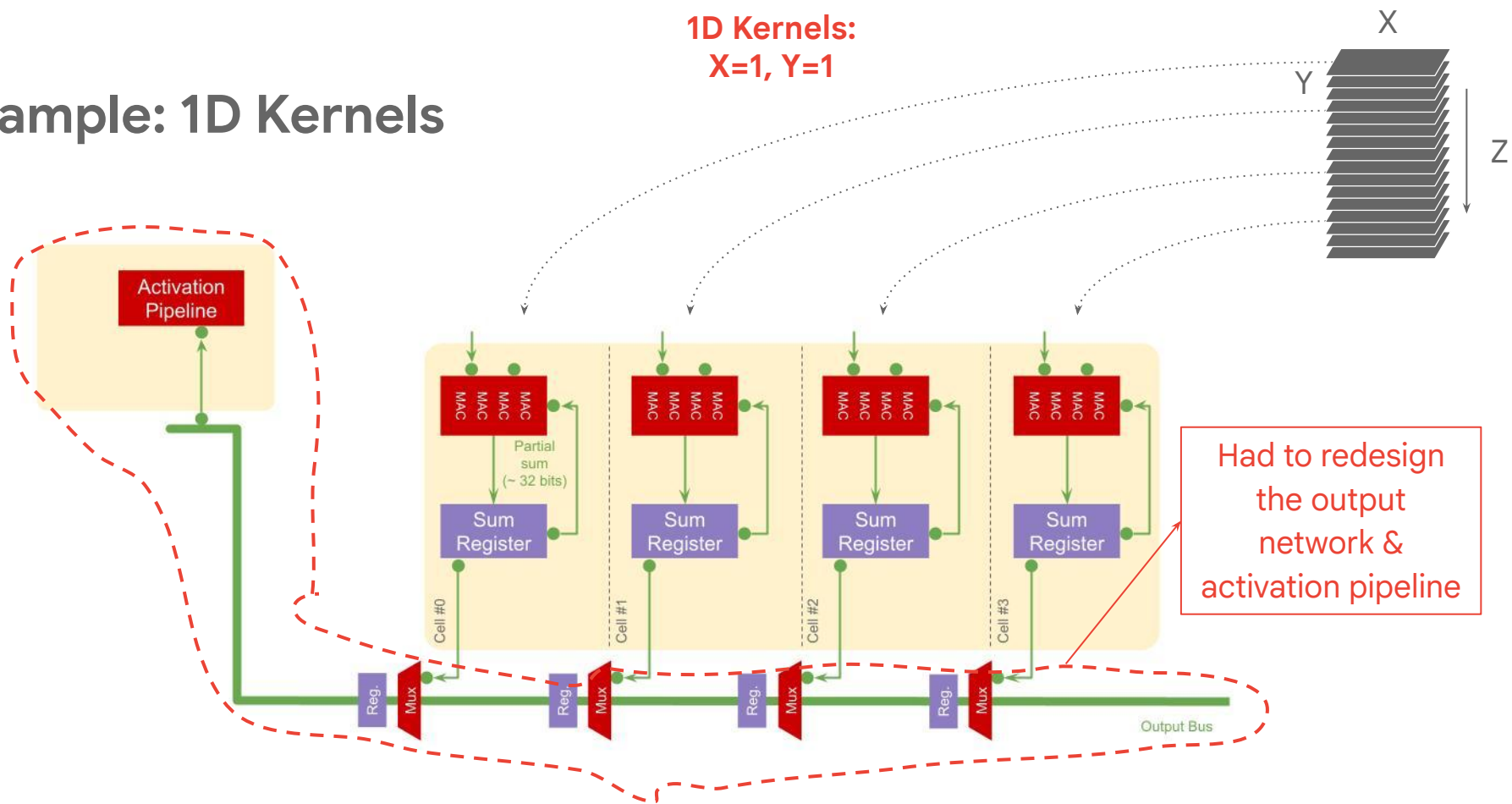
1D Kernels:
 $X=1, Y=1$

Designed for
large $X*Y$



Example: 1D Kernels

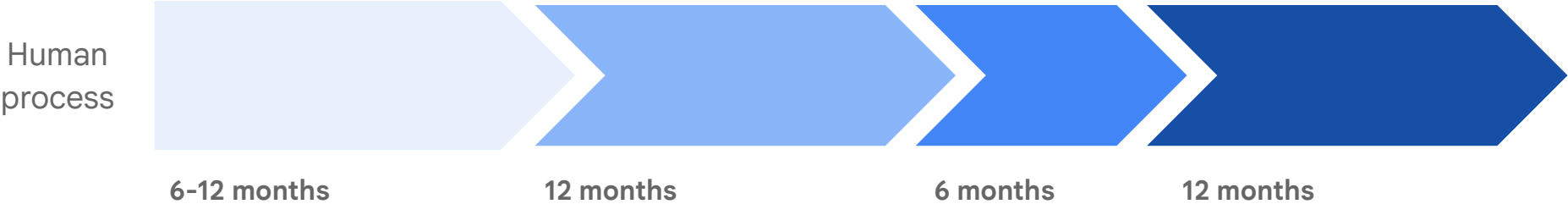
1D Kernels:
 $X=1, Y=1$



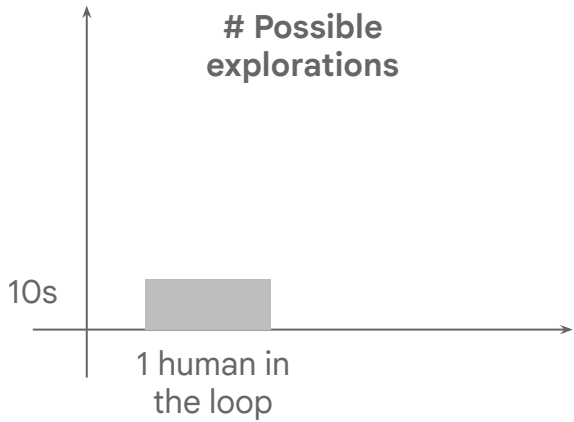
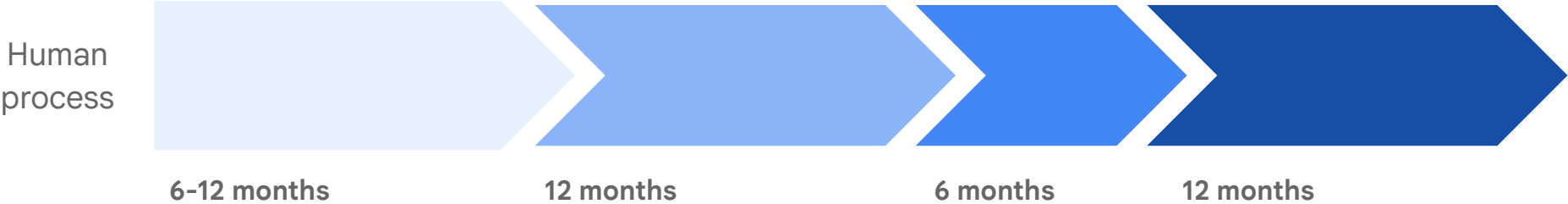
Must design **dramatically** faster



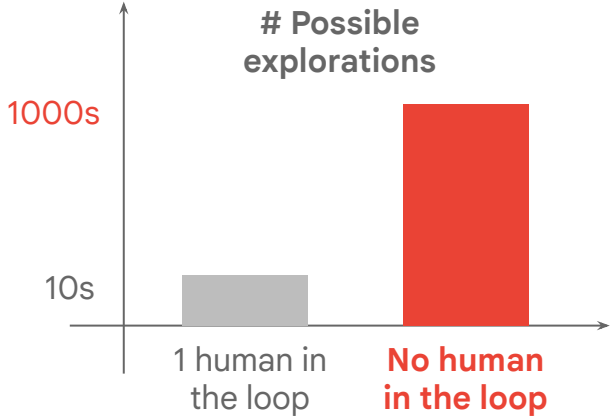
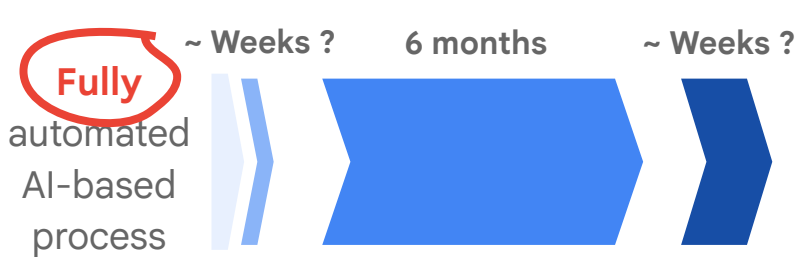
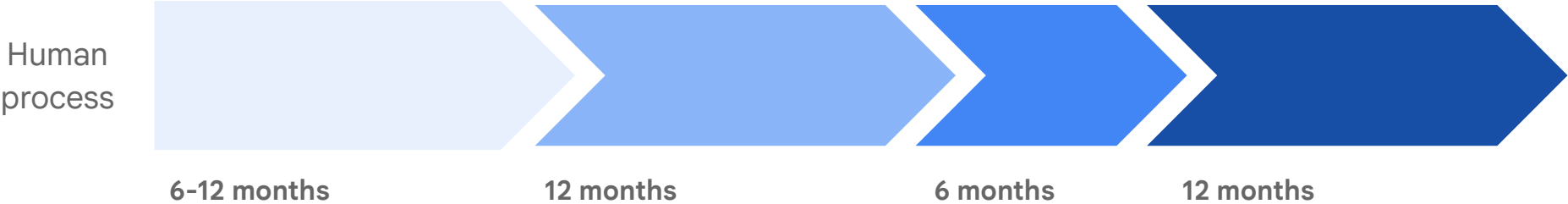
Must design **dramatically** faster



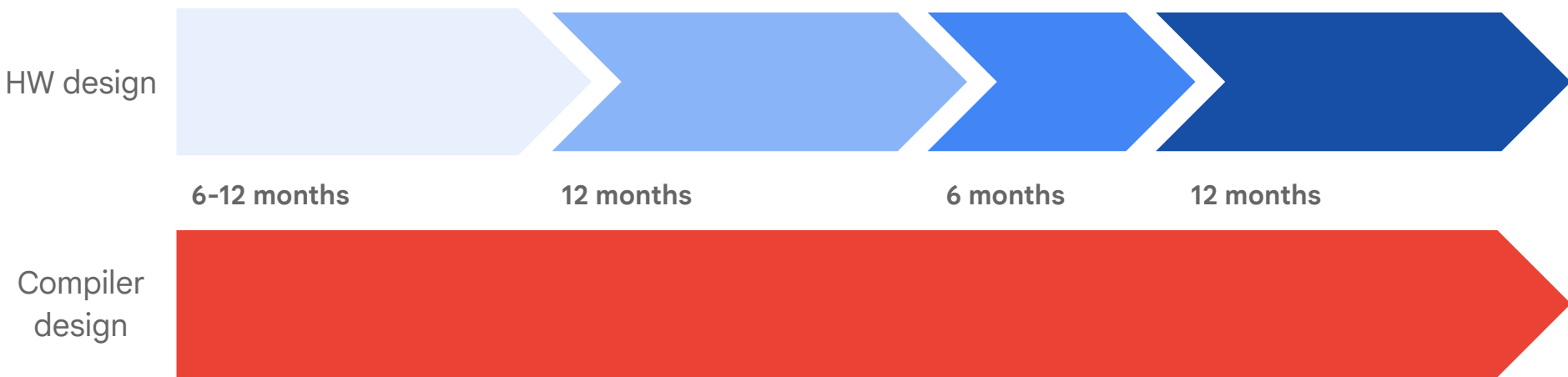
Must design **dramatically** faster



Must design **dramatically** faster



Must also generate the **compiler** !



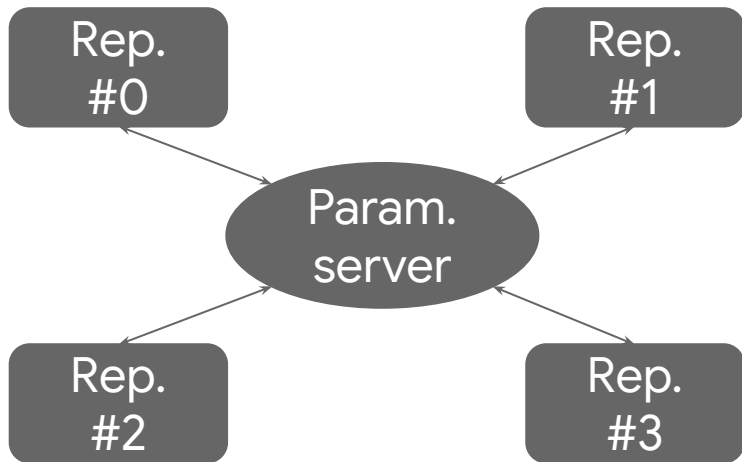
NO TIME TO
UNDERSTAND NEW AI
MODELS IN DEPTH

We don't always understand why our AI models work well...

...and thus how to properly
design the HW for
supporting them

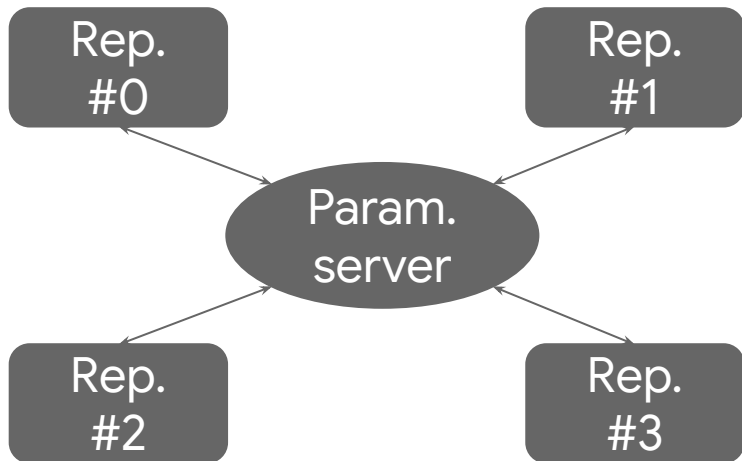


Example: Synchronous replicas



Parameter server →
Async comms & **random order** of
gradients updates in server

Example: Synchronous replicas



Parameter server →
Async comms & **random order** of
gradients updates in server

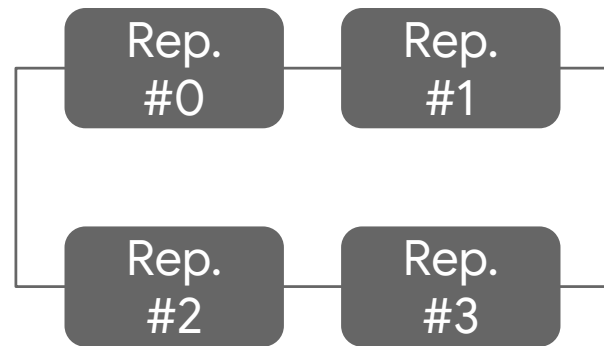
NIPS 2012

Large Scale Distributed Deep Networks

Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen,
Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato,
Andrew Senior, Paul Tucker, Ke Yang, Andrew Y. Ng
{jeff, gcorrado}@google.com
Google Inc., Mountain View, CA

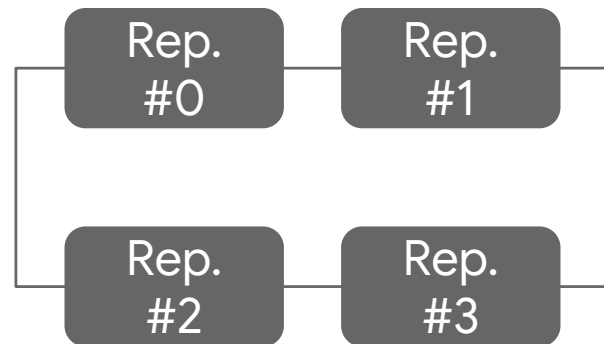
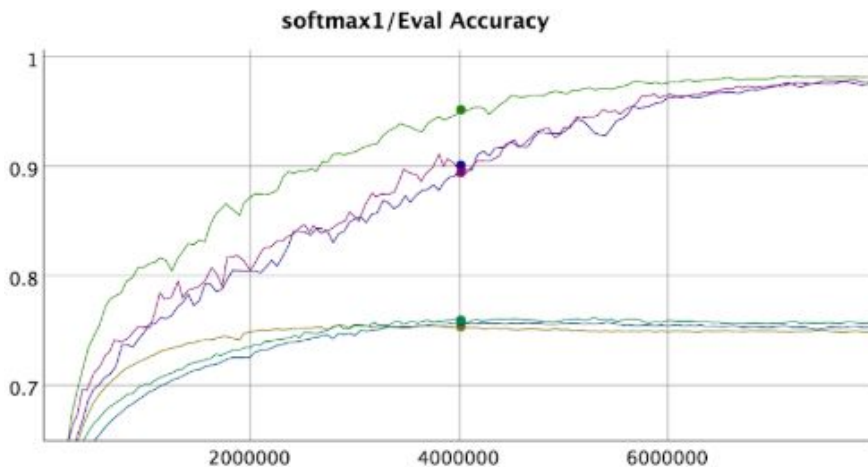
Adagrad was not originally designed to be used with asynchronous SGD, and neither method is typically applied to nonconvex problems. It is surprising, therefore, that they work so well together, and on highly nonlinear deep networks. We conjecture that Adagrad automatically stabilizes volatile parameters in the face of the flurry of asynchronous updates, and naturally adjusts learning rates to the demands of different layers in the deep network.

Example: Synchronous replicas



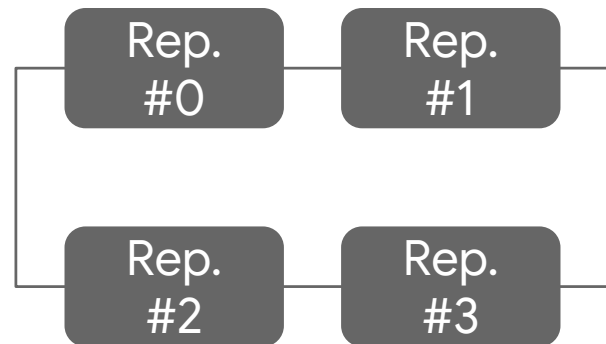
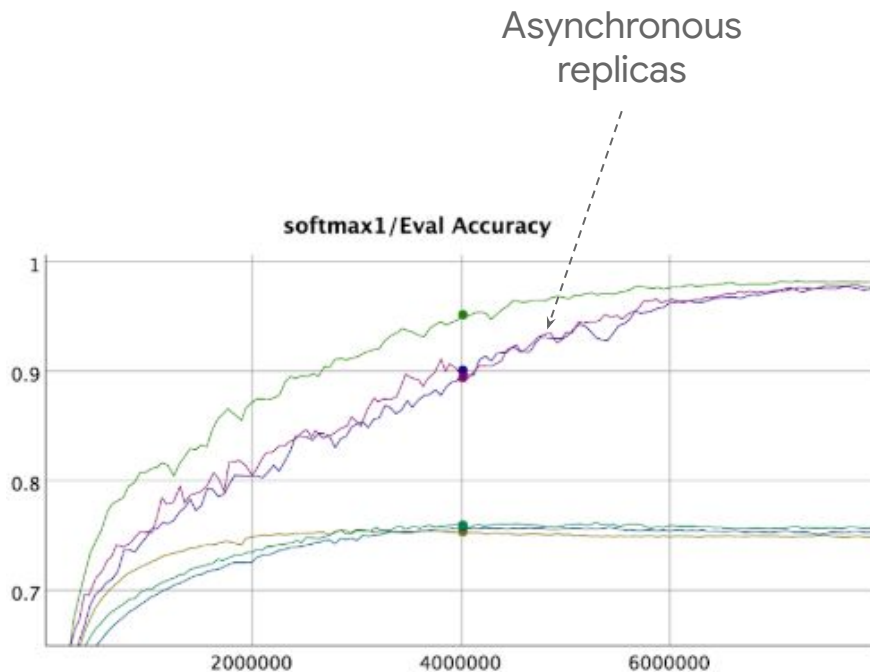
Ring topology →
Synchronous comms, and **fixed order**
of gradients updates

Example: Synchronous replicas



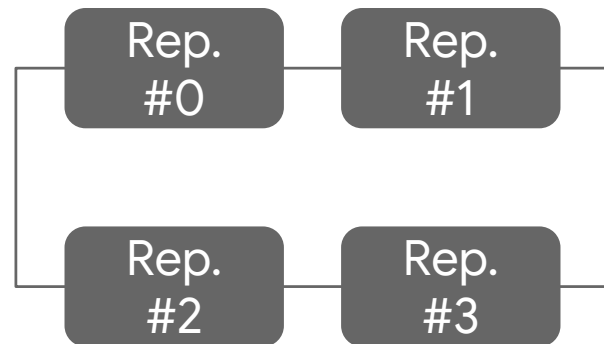
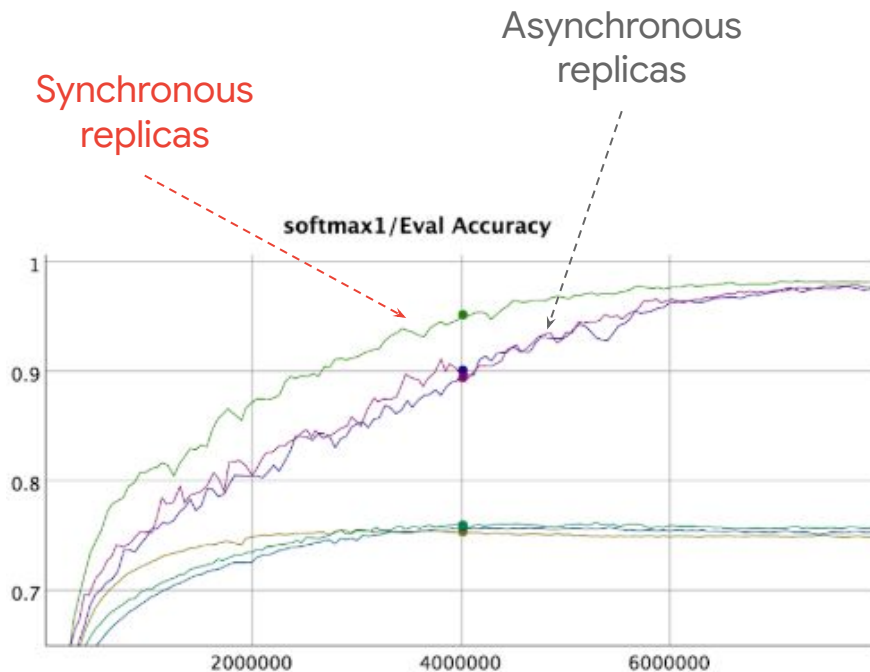
Ring topology →
Synchronous comms, and **fixed order**
of gradients updates

Example: Synchronous replicas



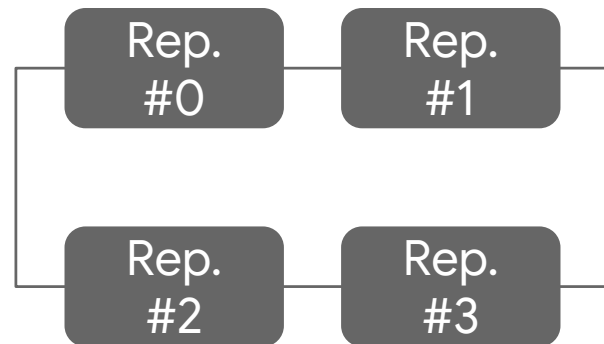
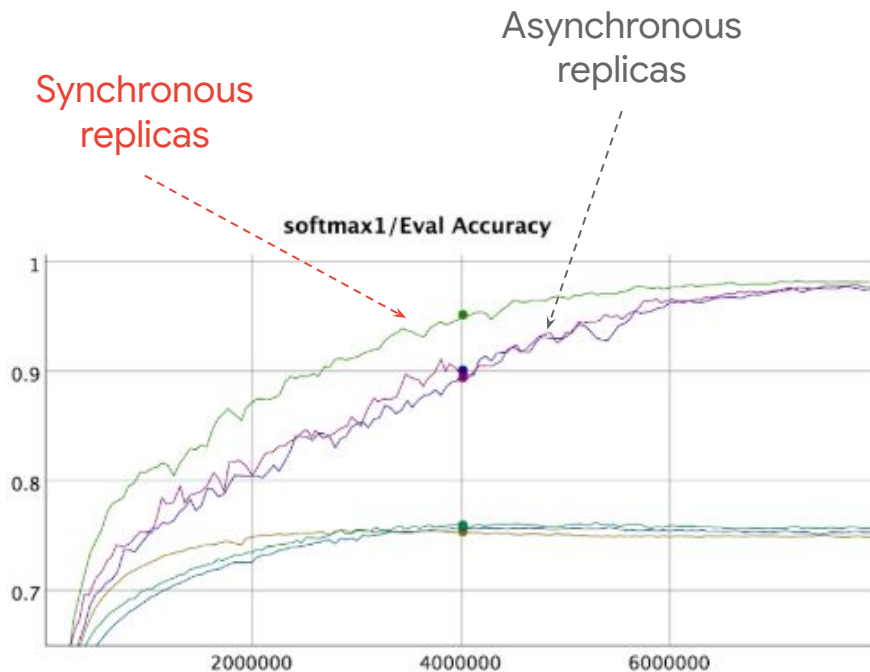
Ring topology →
Synchronous comms, and **fixed order**
of gradients updates

Example: Synchronous replicas



Ring topology →
Synchronous comms, and **fixed order**
of gradients updates

Example: Synchronous replicas



Ring topology →
Synchronous comms, and **fixed order**
of gradients updates

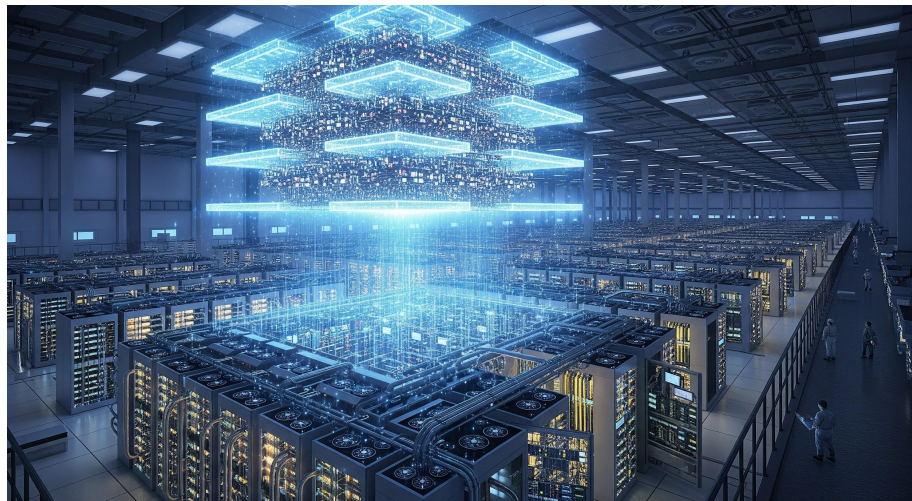
Conclusions

- Synchrony is actually great!

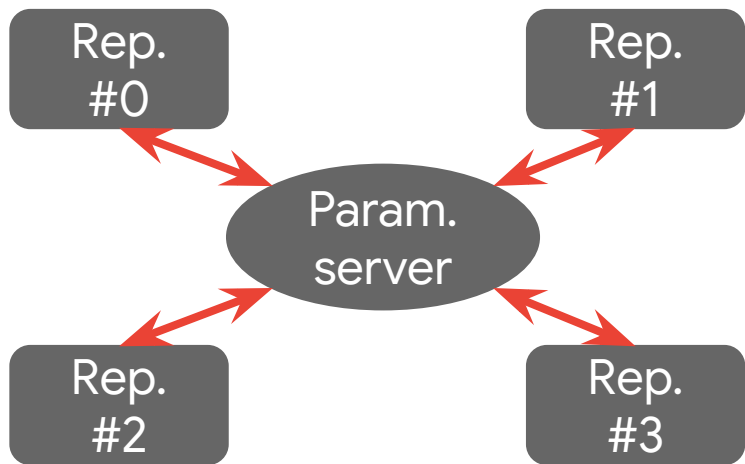
NO TIME TO VALIDATE
NEW HARDWARE IDEAS

We are lacking tools to design AI hardware **quickly**

- **We have to design super fast**
- Simulation is extremely slow
- AI training time can be in weeks/months (over many chips)

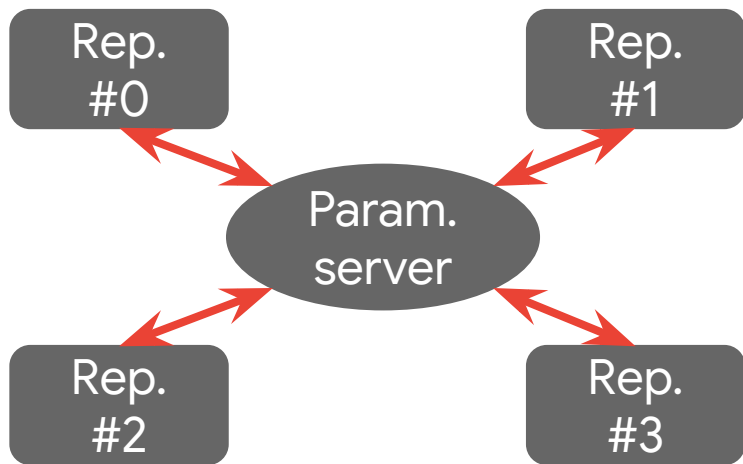


Example: BFloat16 (BrainFloat16)



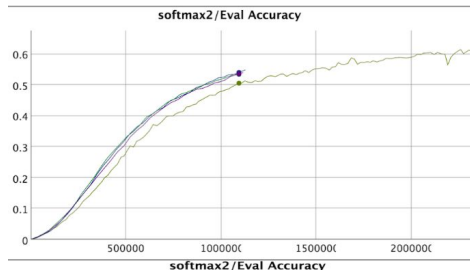
 Mantissa truncated from 23 to 7 bits to reduce network traffic (BF16)

Example: BFloat16 (BrainFloat16)



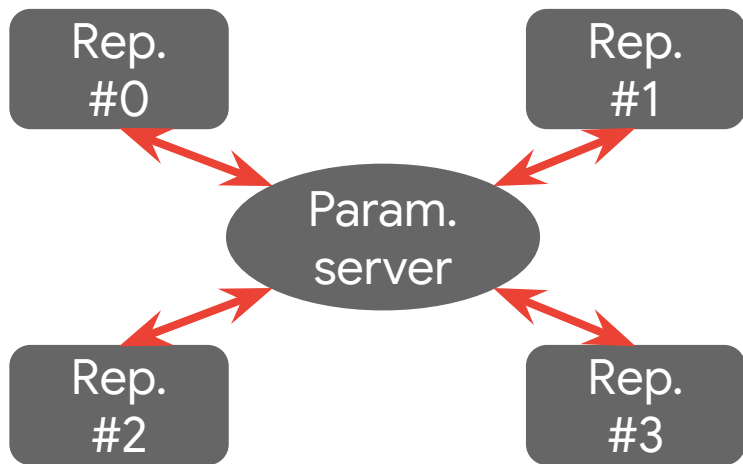
↔ Mantissa truncated from 23 to 7 bits to reduce network traffic (BF16)

Imagenet Results So Far



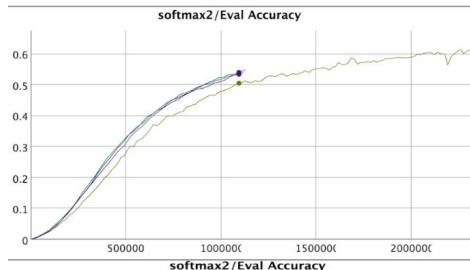
- fp runs:
 - a. better than DB
 - b. 5x slower

Example: BFloat16 (BrainFloat16)



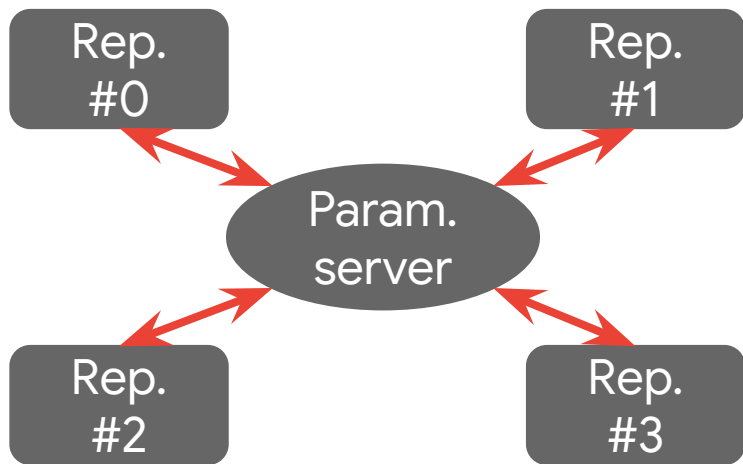
 Mantissa truncated from 23 to 7 bits to reduce network traffic (BF16)

Imagenet Results So Far



- fp runs:
 - a. better than DB
 - b. 5x slower

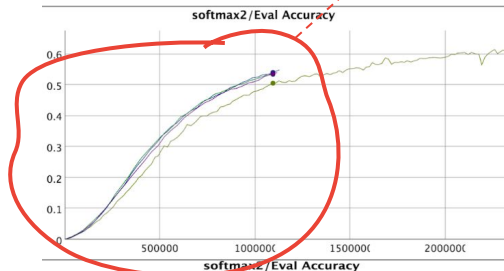
Example: BFloat16 (BrainFloat16)



↔ Mantissa truncated from 23 to 7 bits to reduce network traffic (BF16)

Was tracking, but **training wasn't done when we taped out !!**

Imagenet Results So Far



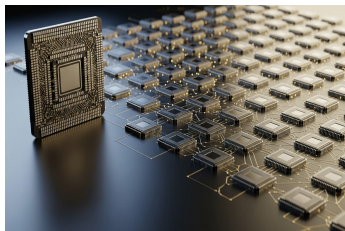
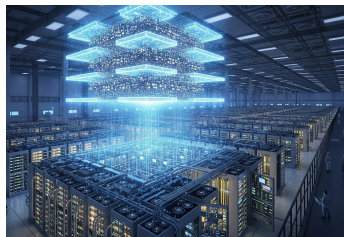
- fp runs:
 - a. better than DB
 - b. 5x slower

1095770:
imagenet_112514.pyplan/db_eva_test_imgnet/db_eva_test_imgnet: 0.5061
imagenet_112514.pyplan/pdb_eva_test_imgnet_a_6_31_20_m_6_31_6/pdb_eva_test_imgnet_a_6_31_20_m_6_31_6: 0.5409
imagenet_112514.pyplan/pdb_eva_test_imgnet_a_6_31_20_m_6_31_7/pdb_eva_test_imgnet_a_6_31_20_m_6_31_7: 0.5348
imagenet_112514.pyplan/pdb_eva_test_imgnet_a_6_31_21_m_6_31_6/pdb_eva_test_imgnet_a_6_31_21_m_6_31_6: 0.5361

MOREOVER...

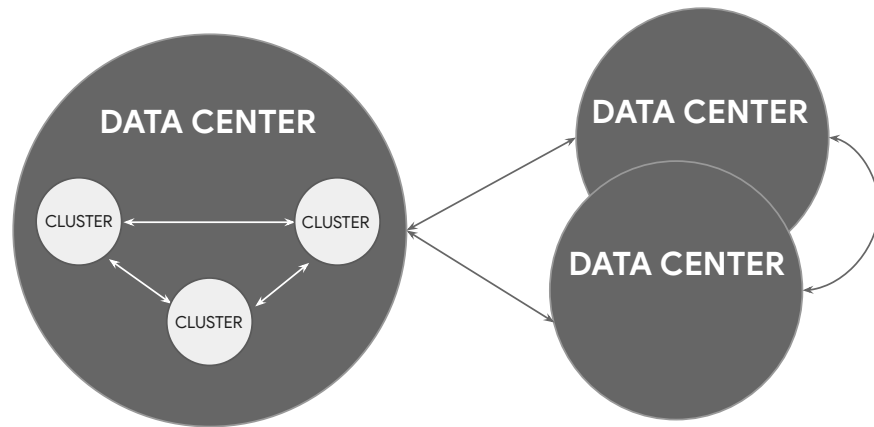
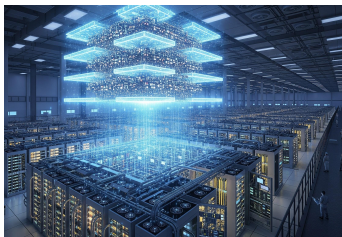
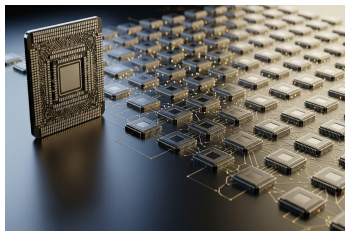
GOES WELL BEYOND CHIP DESIGN

Chips challenges
can be dwarfed by
systems challenges



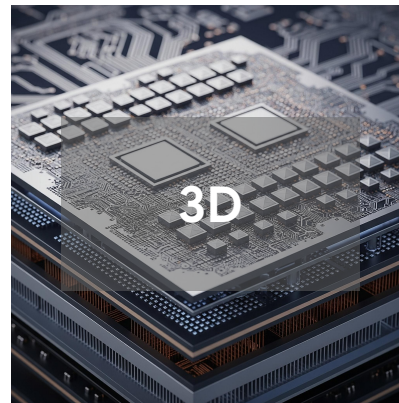
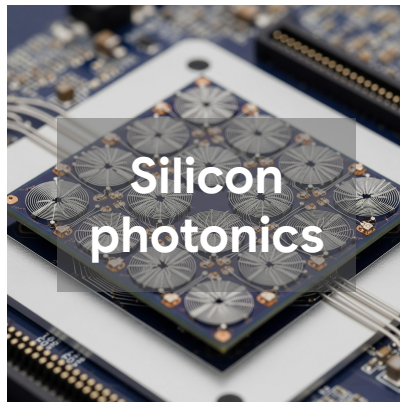
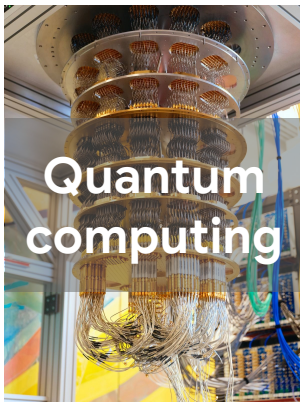
GOES WELL BEYOND CHIP DESIGN

Chips challenges
can be dwarfed by
systems challenges



Some versions of Gemini are trained
across **multiple data centers**

NO TIME TO EXPLORE THE MANY TECHNOLOGY OPPORTUNITIES



THANK YOU !