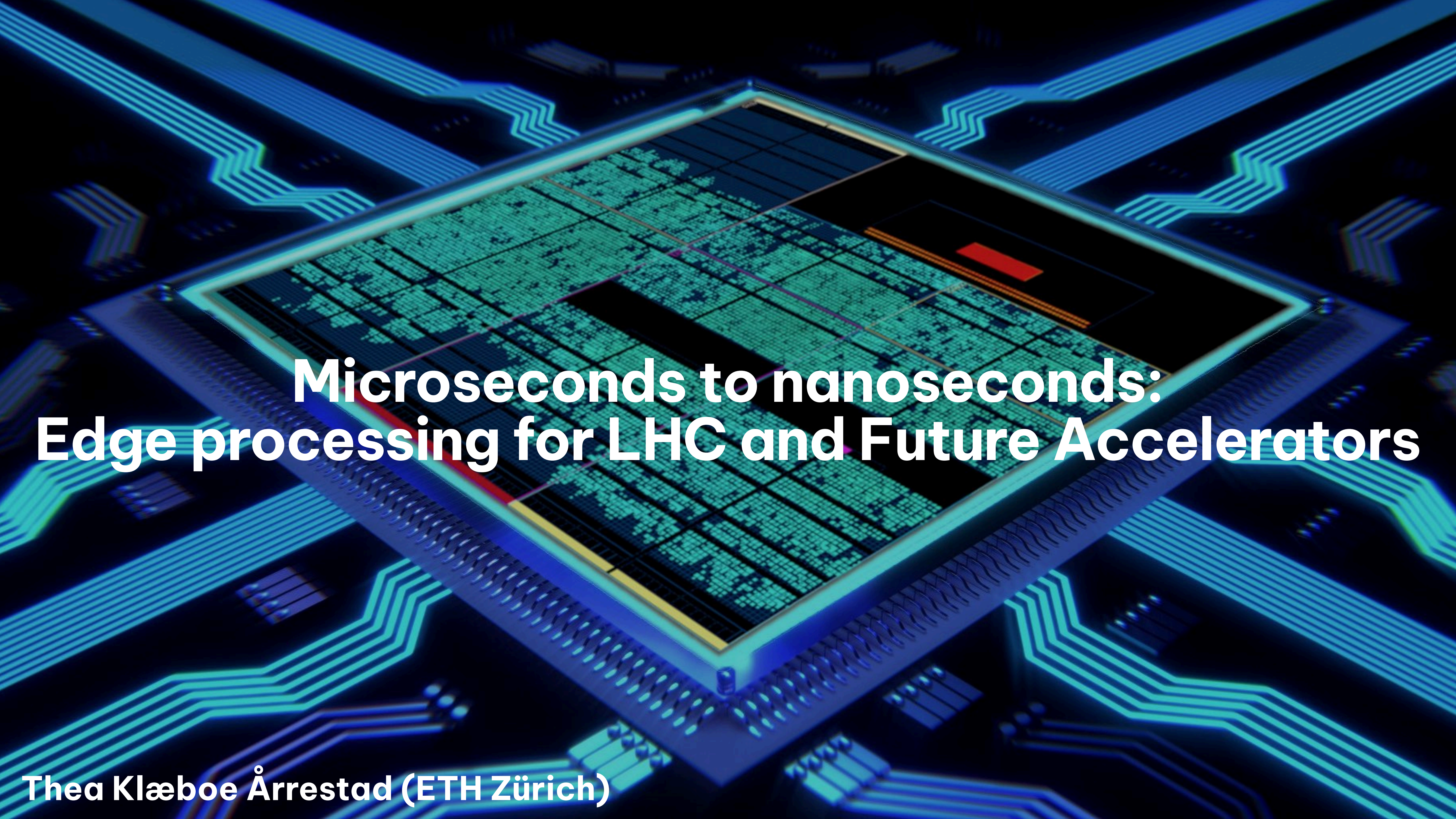
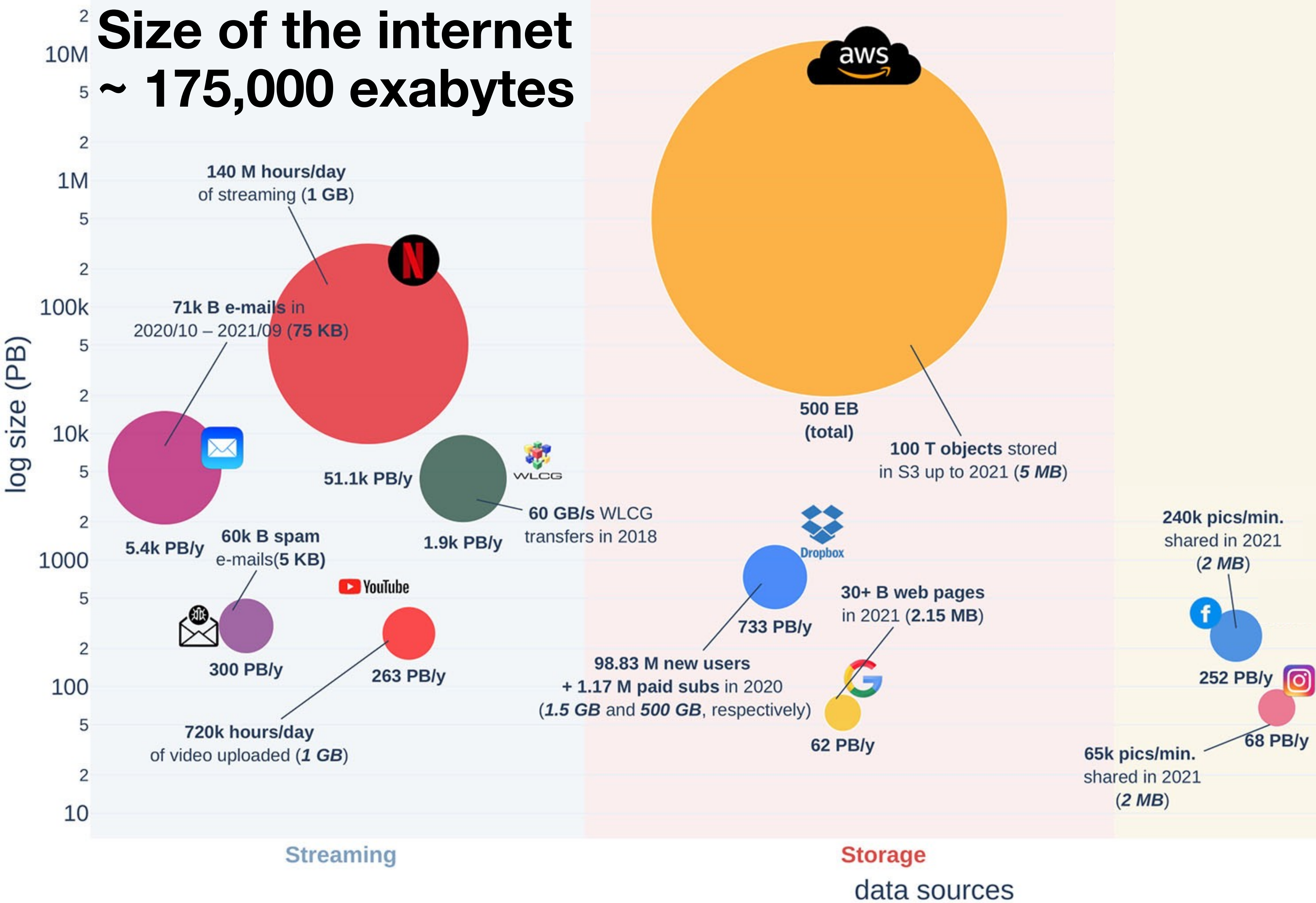


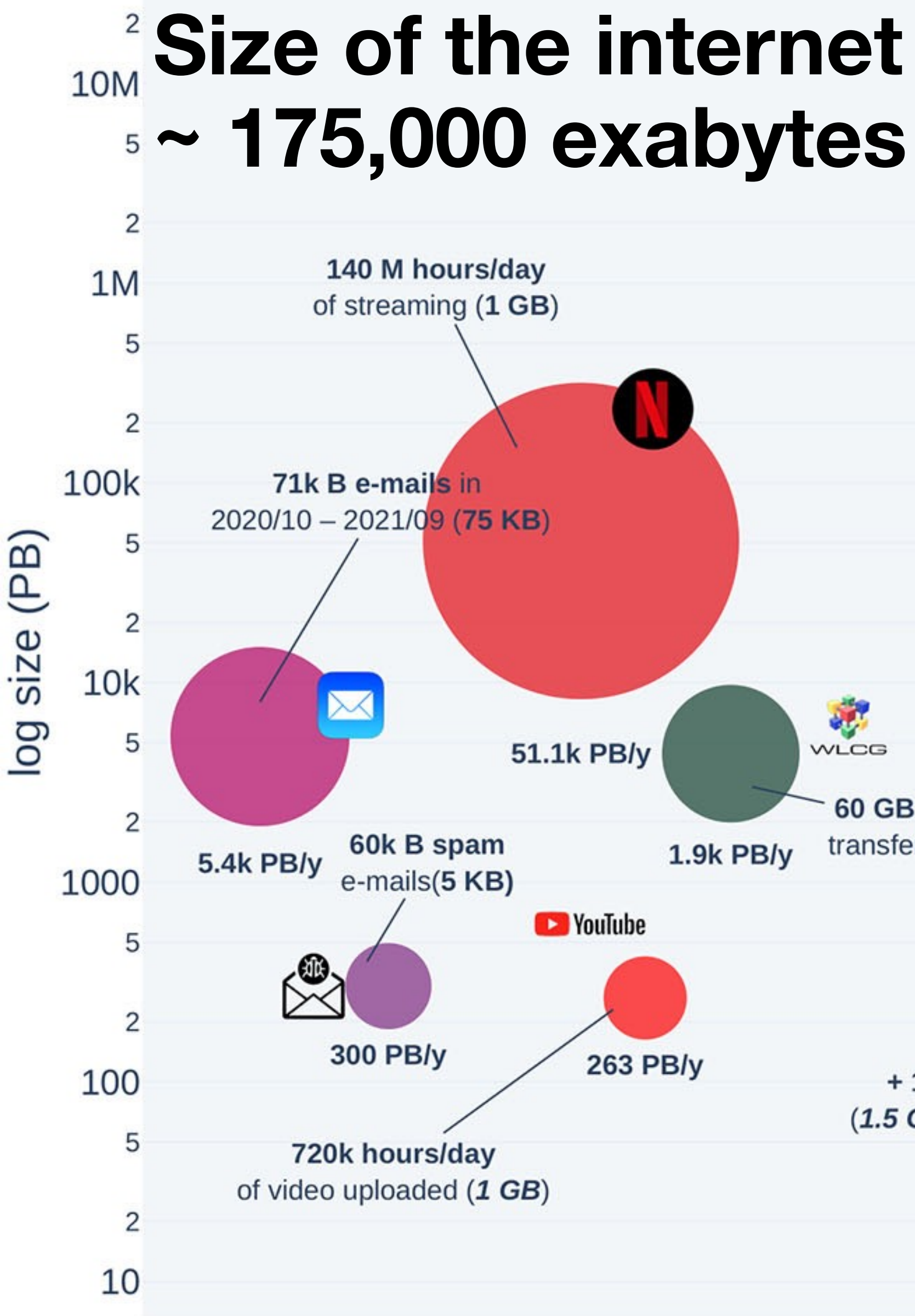


Microseconds to nanoseconds: Edge processing for LHC and Future Accelerators

Theo Klæboe Årrestad (ETH Zürich)



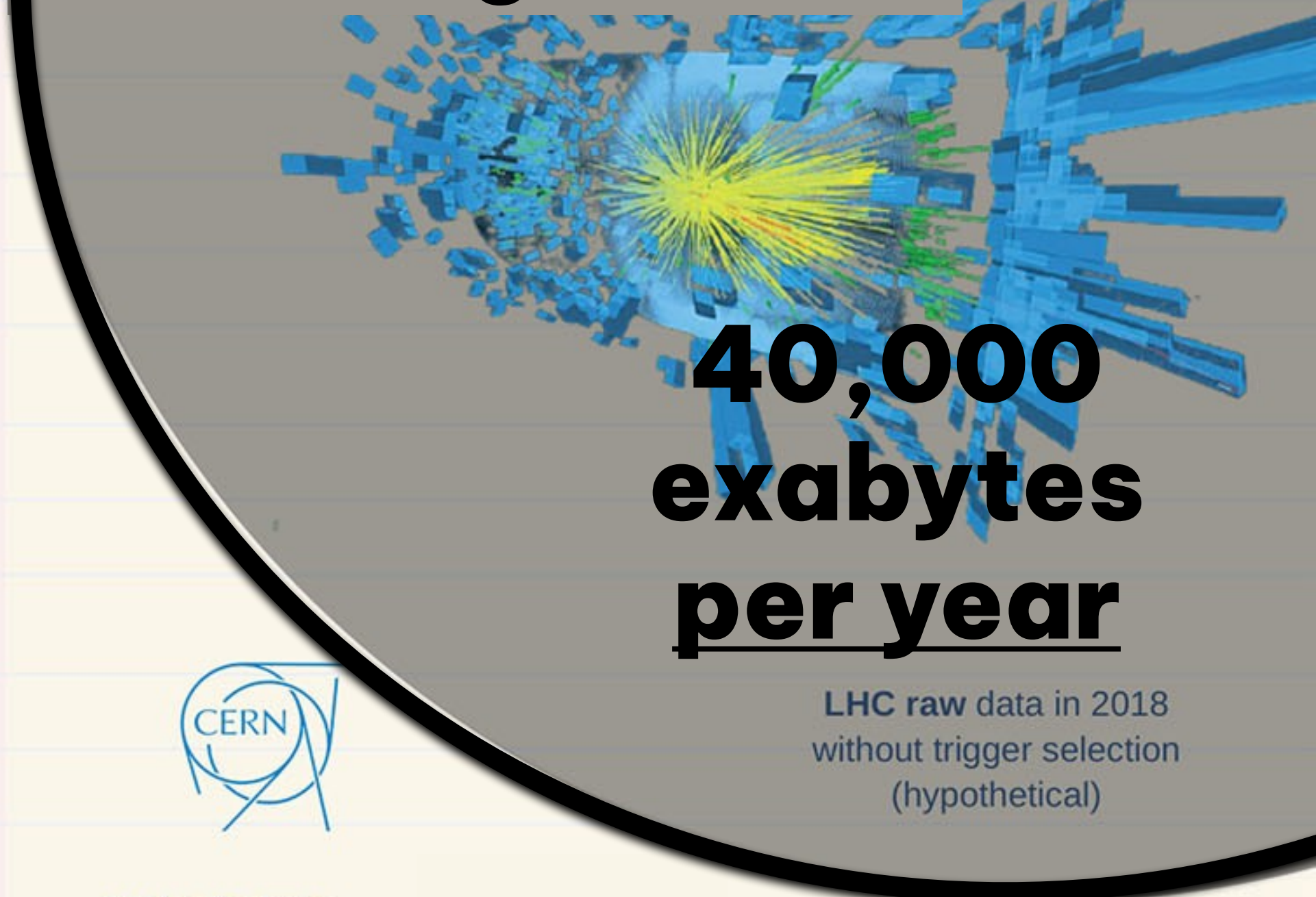
Size of the internet ~ 175,000 exabytes



Streaming

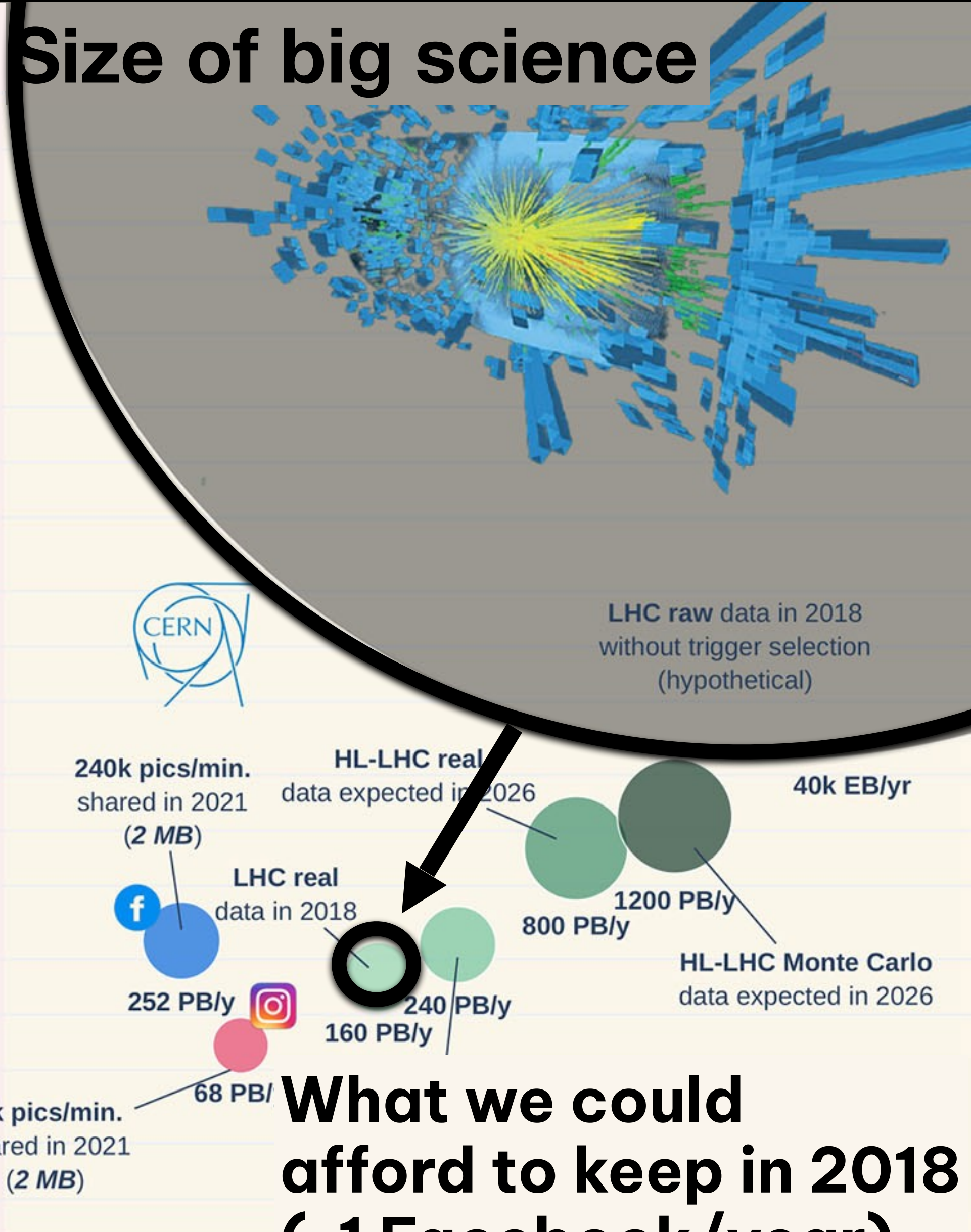
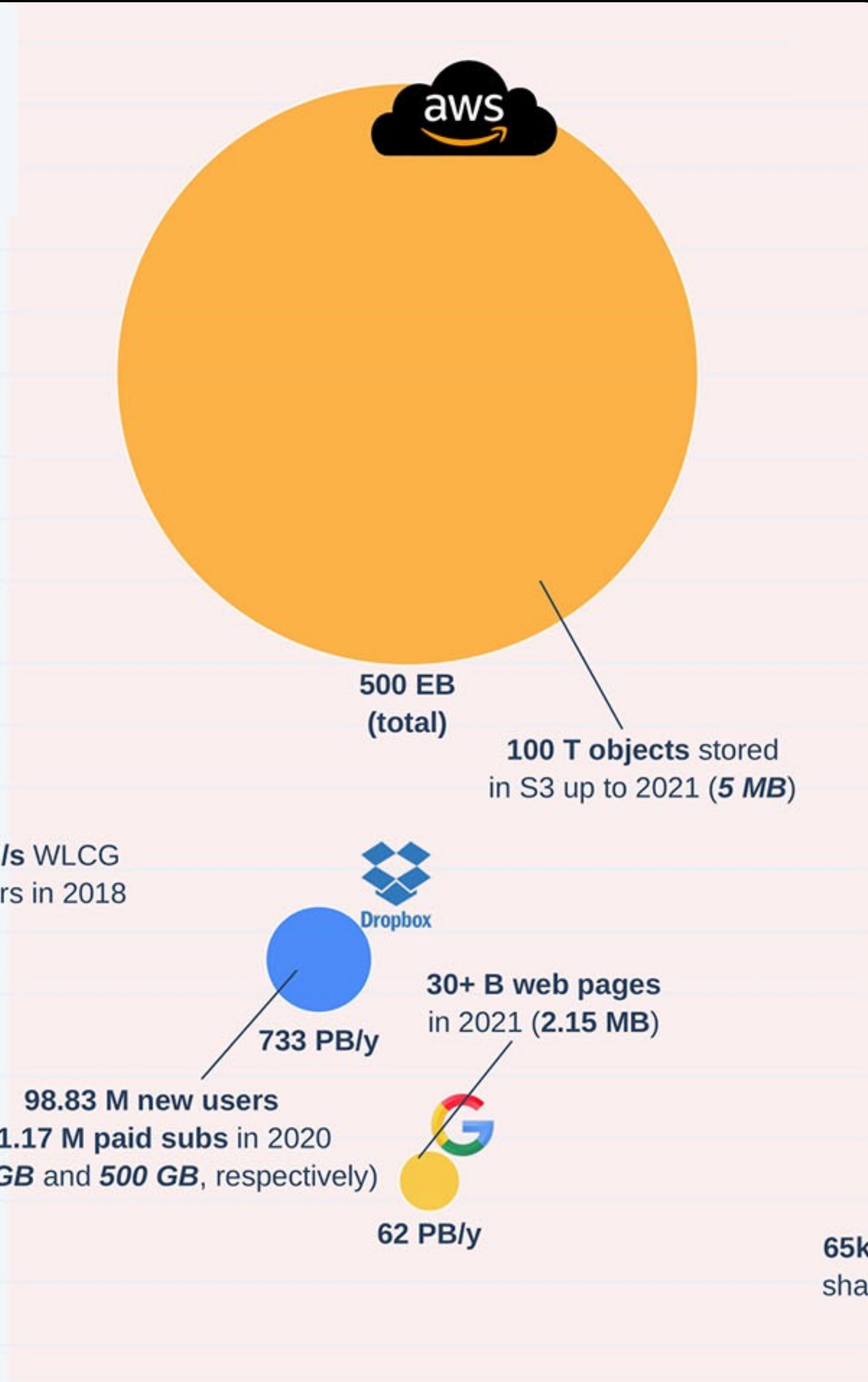
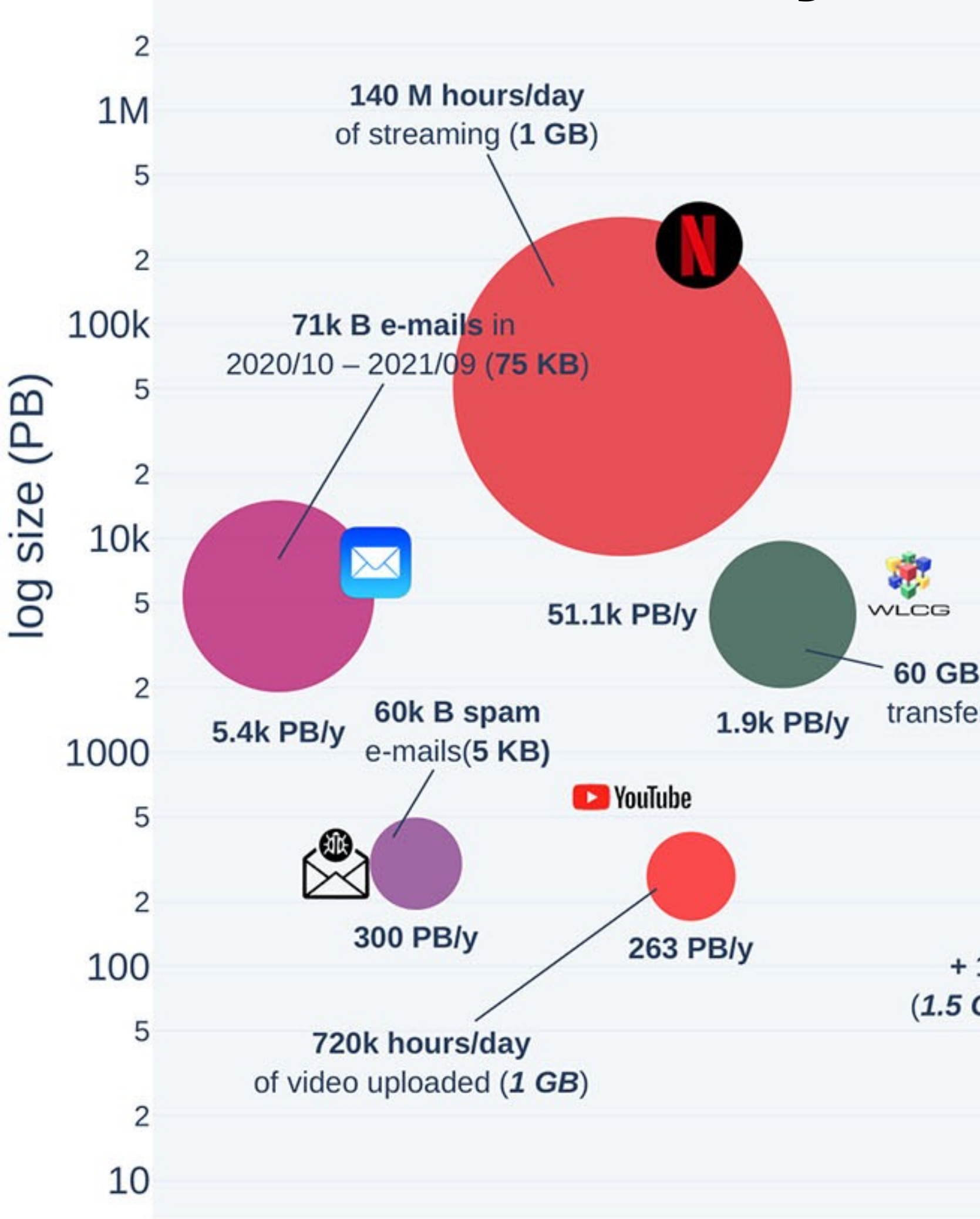
Storage
data sources

Size of big science



Production

Size of the internet ~ 175,000 exabytes

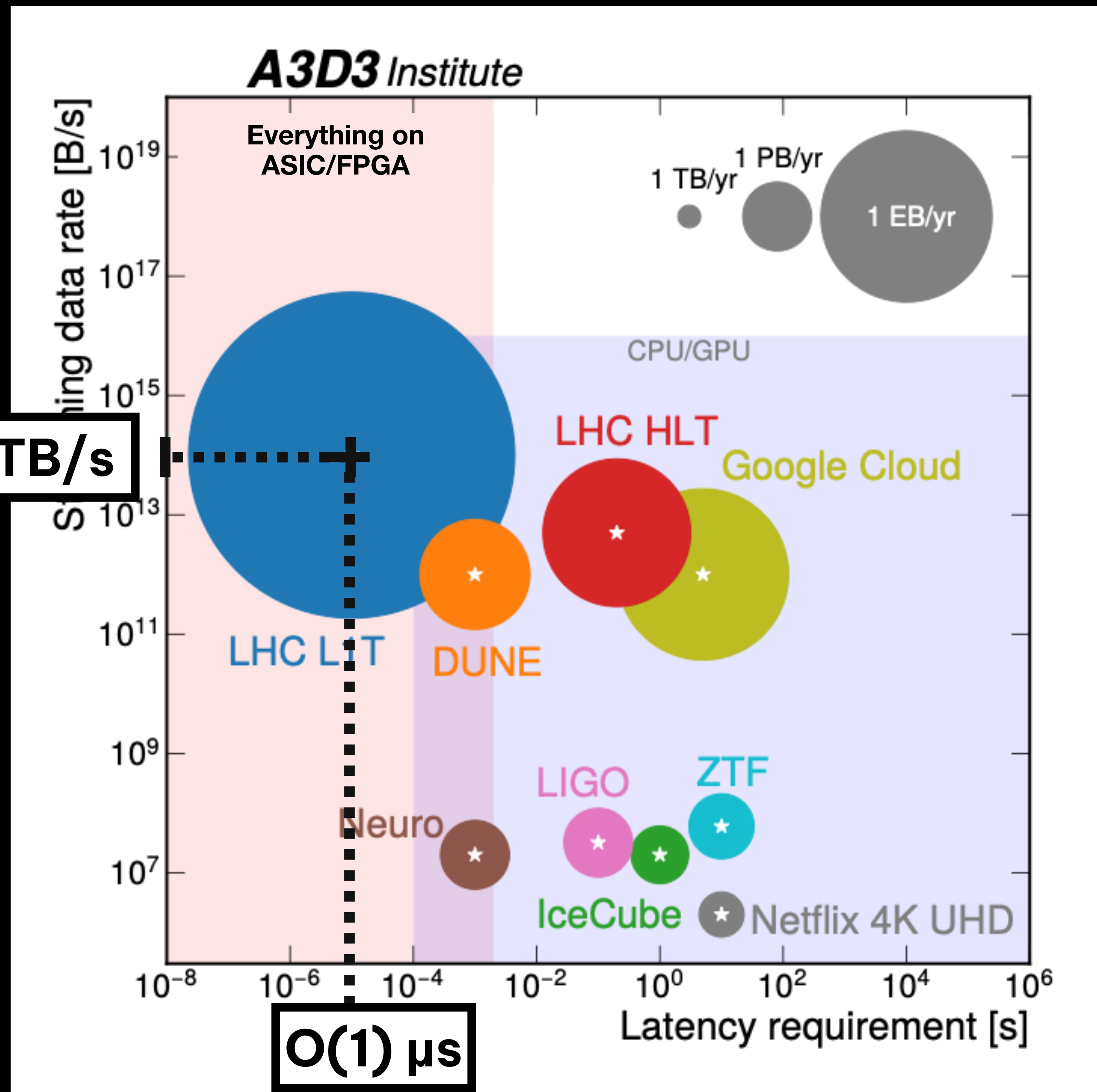


What we could afford to keep in 2018 (~1 Facebook/year)

And how fast
do we need to
decide what to
keep?

O(100) TB/s

How much
data goes
through system



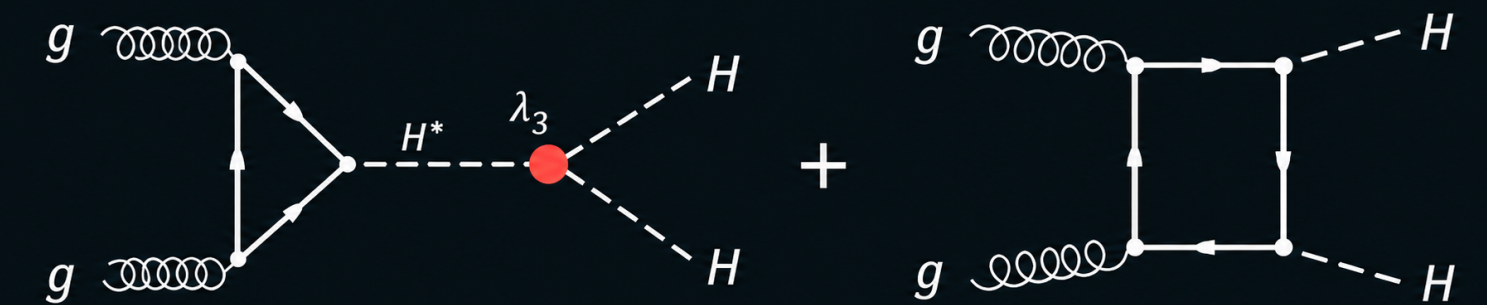
O(1) μs

Time to process

And why bother?

The physics we want to do, require that we sift through **one hundred thousand billion collisions** to measure processes occurring 1 in $\sim 10^{13}$ (or less!)

Example: Di-Higgs production



Triangle diagram
sensitive to the Higgs
self-coupling λ_3

Box diagram
interferes destructively
with the triangle

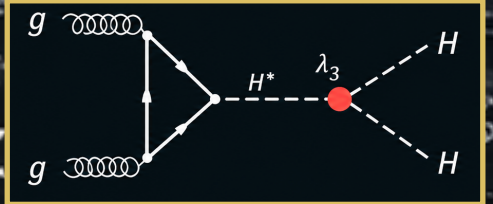
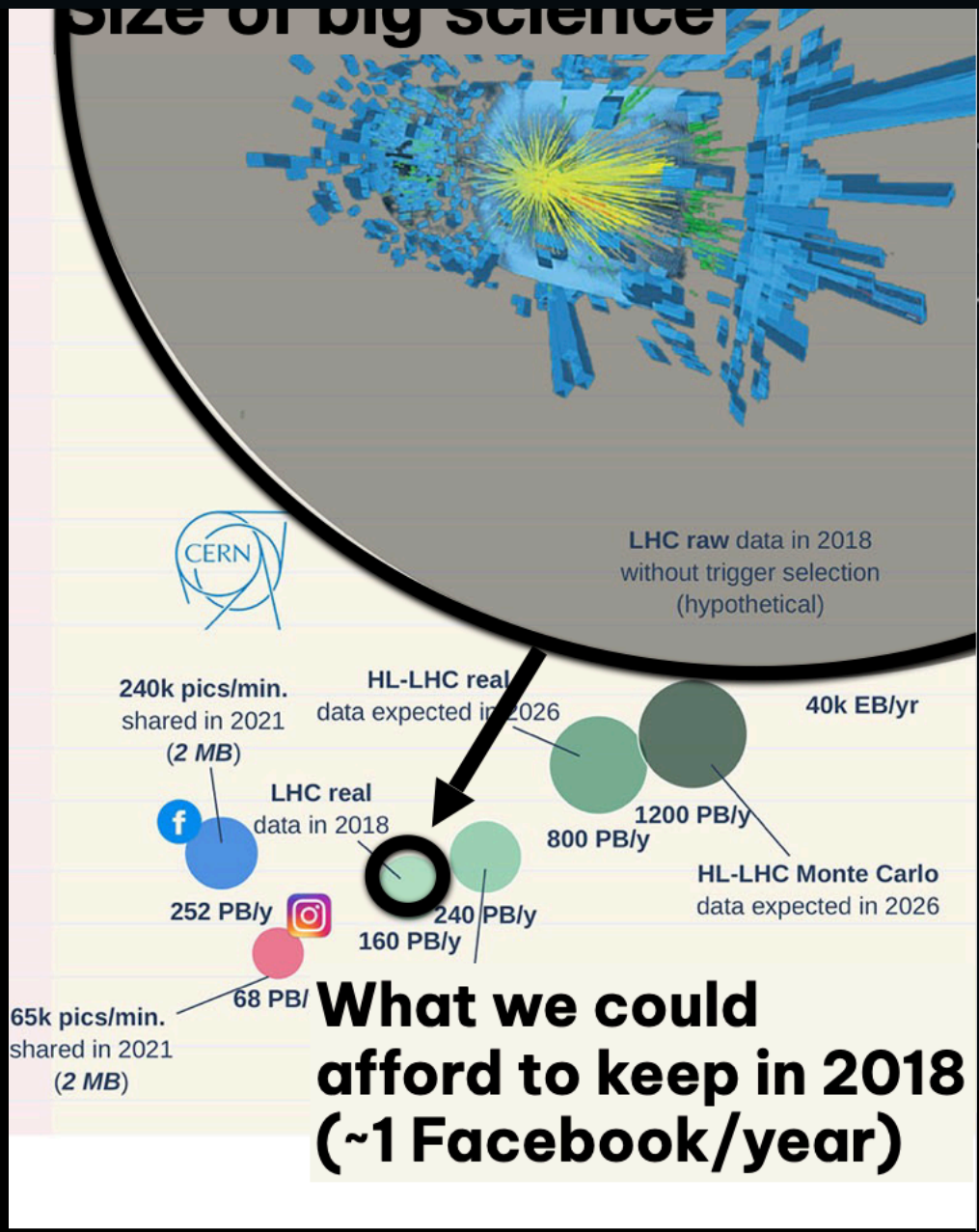
Thanks to quantum mechanics, these diagrams even interfere destructively, making the process even rarer!

We want to measure it so accurately that there is only a **1 in a 780 billion chance (7σ)** that it is a statistical fluke

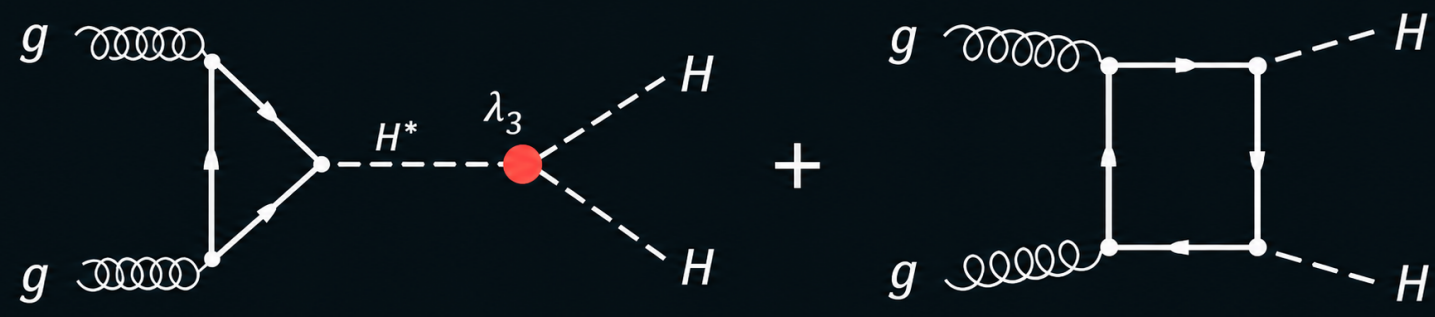
And why bother?

The physics we want to do, require that we sift through **one hundred thousand billion collisions** to measure processes occurring 1 in $\sim 10^{13}$ (or less!)

Aside: That's only this **little** box, I'll get to the big one.



Example: Di-Higgs production



Triangle diagram sensitive to the Higgs self-coupling λ_3

Box diagram interferes destructively with the triangle

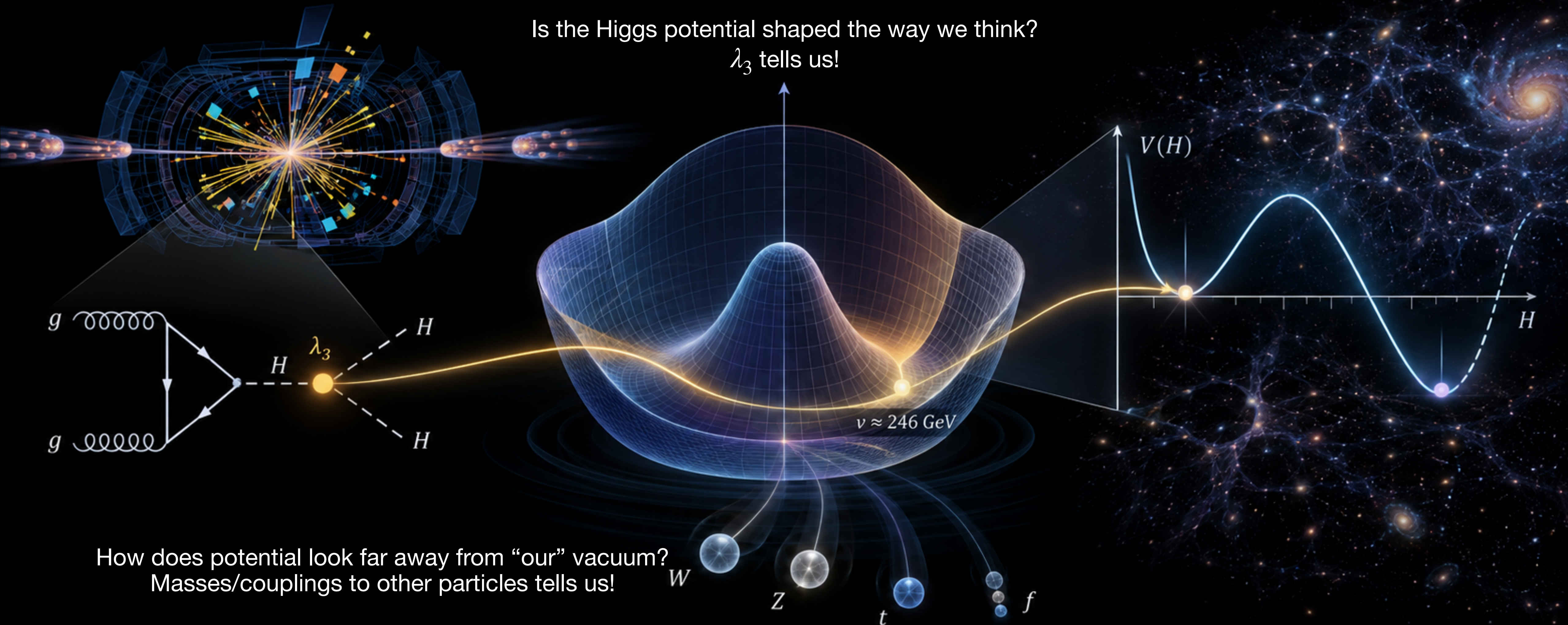
Thanks to quantum mechanics, these diagrams even interfere destructively, making the process even rarer!

We want to measure it so accurately that there is only a **1 in a 780 billion chance (7σ)** that it is a statistical fluke

And why bother, really?

Is our vacuum ultimately stable or will it decay through quantum tunneling?

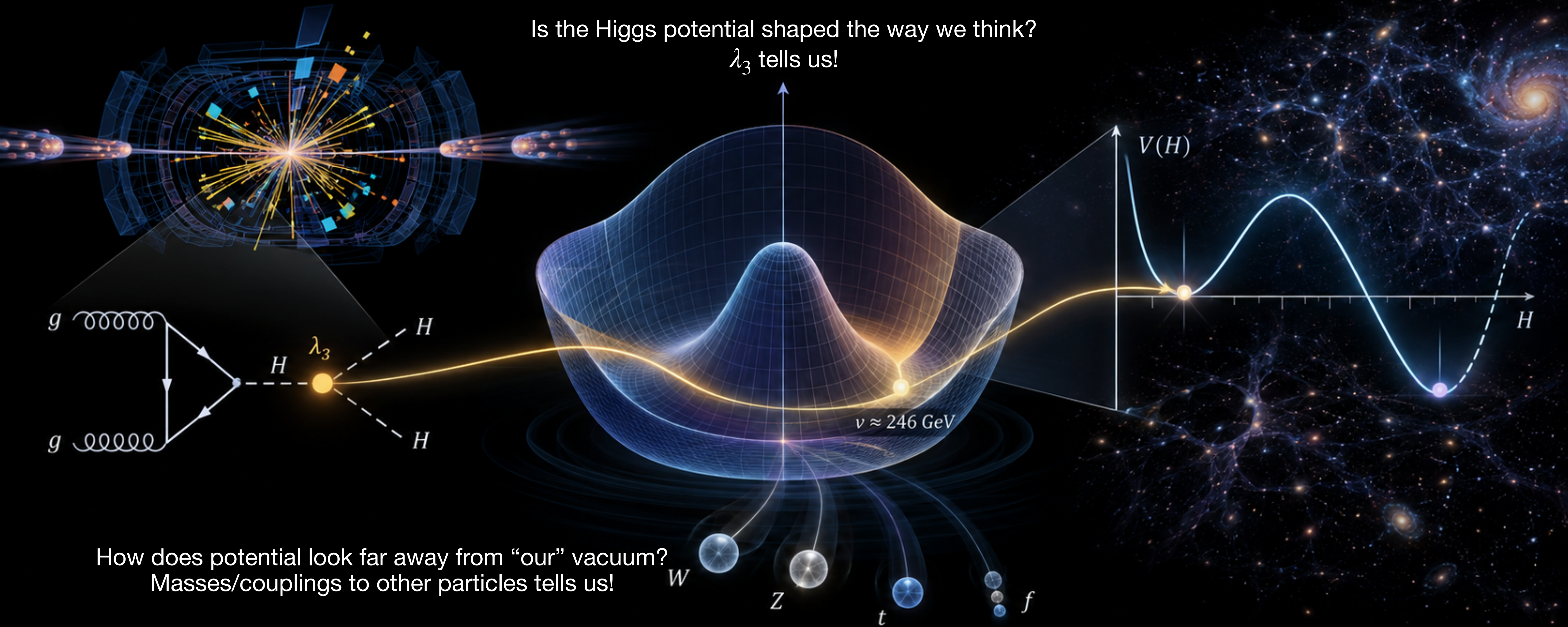
Is the Higgs potential shaped the way we think?
 λ_3 tells us!



How does potential look far away from “our” vacuum?
Masses/couplings to other particles tells us!

And why bother, really?

Is our vacuum ultimately stable or will it decay through quantum tunneling?

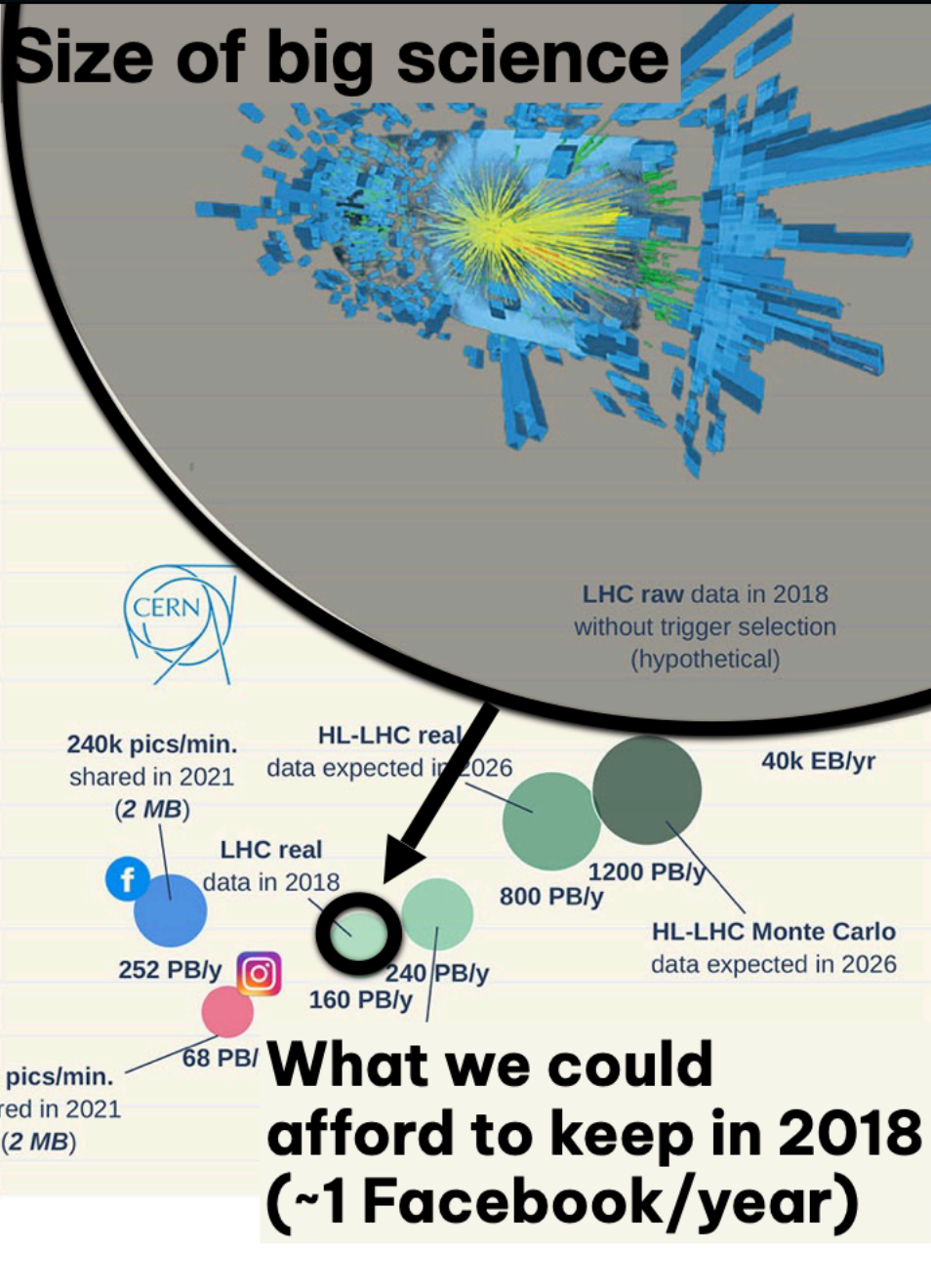


How does potential look far away from “our” vacuum?
Masses/couplings to other particles tells us!

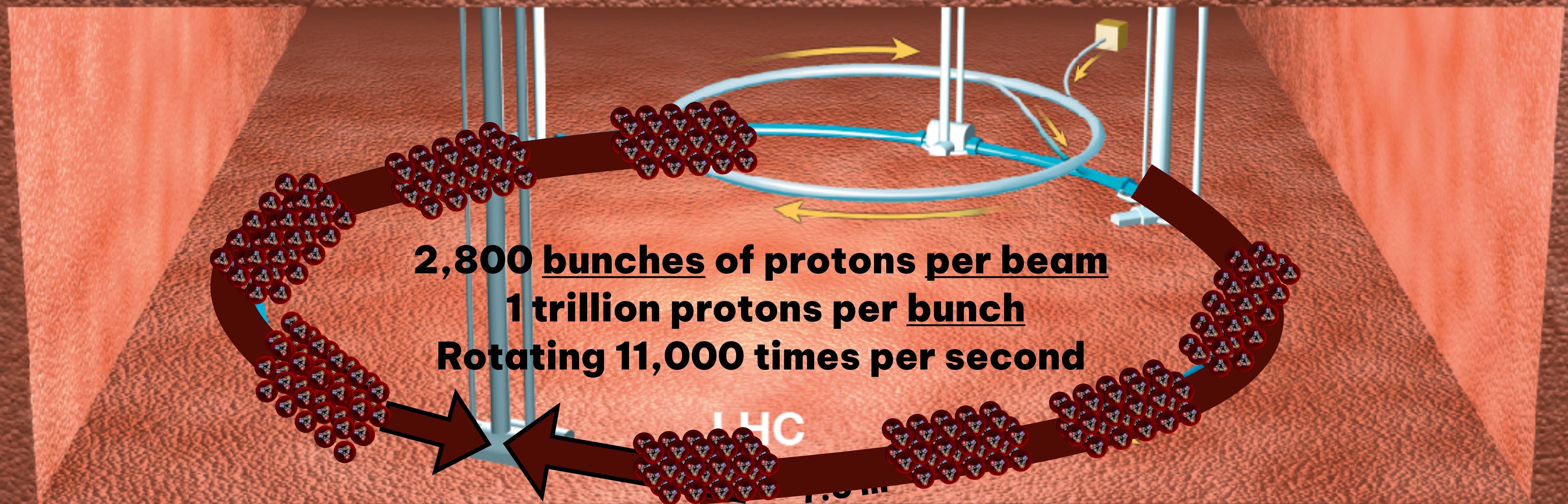
Will our universe ultimately boil* because bubbles of a different vacuum nucleate and spread through our universe at the speed of light?

[*] at a timescale longer than the current age of the universe

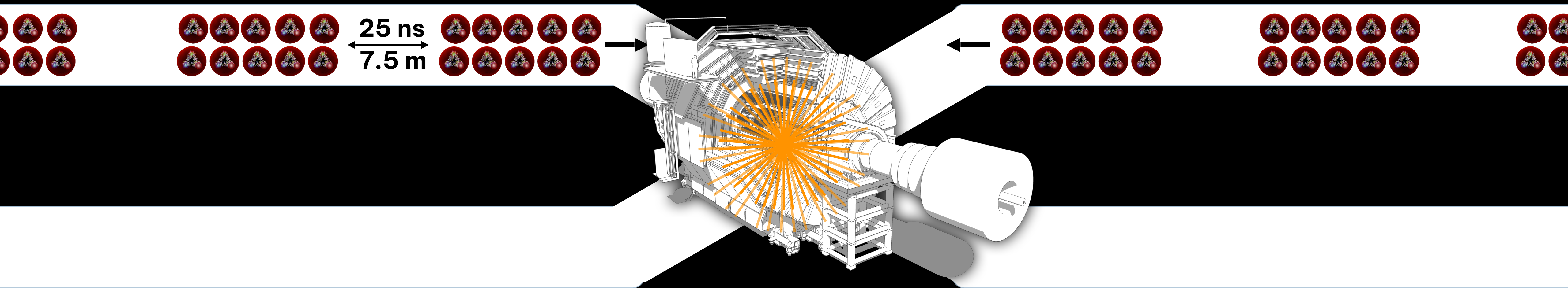
I said I'd tell you
about the big one.



The Big One



**These bunches “collide” every 25 ns (=40 MHz)
(densely packed, want much data!)**

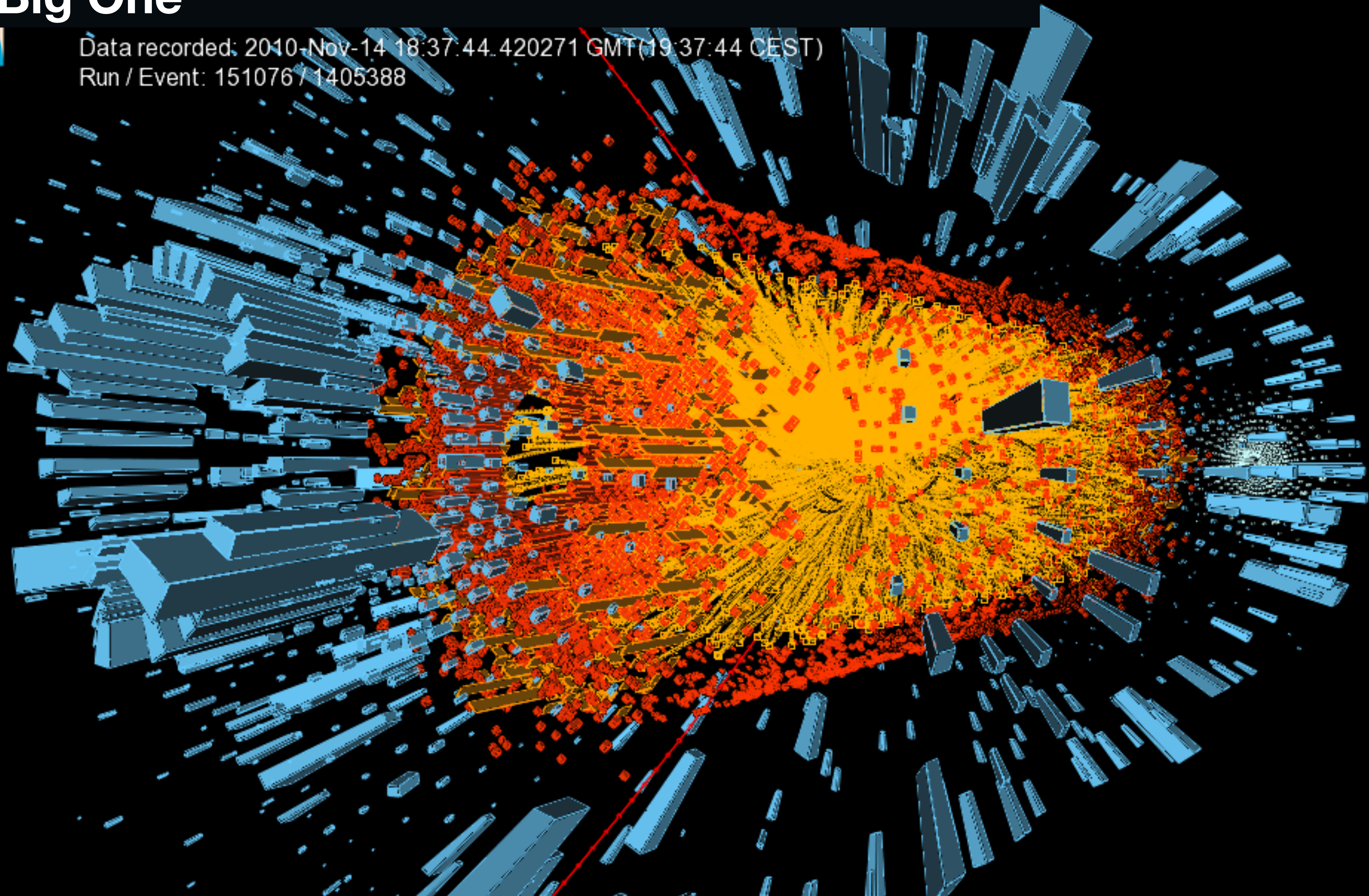


**~Out of 10^{12} protons, only 60 collisions per crossing
Interaction is extremely rare**

The Big One



Data recorded: 2010-Nov-14 18:37:44.420271 GMT(19:37:44 CEST)
Run / Event: 151076 / 1405388





CMS Experiment at the LHC, CERN

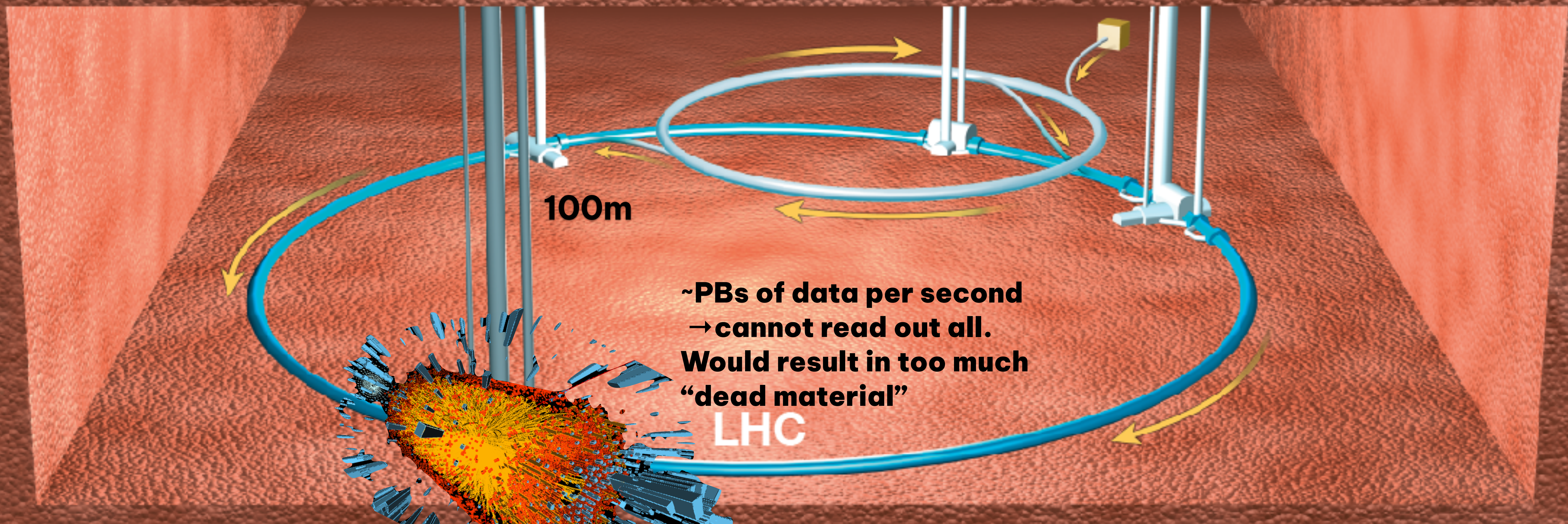
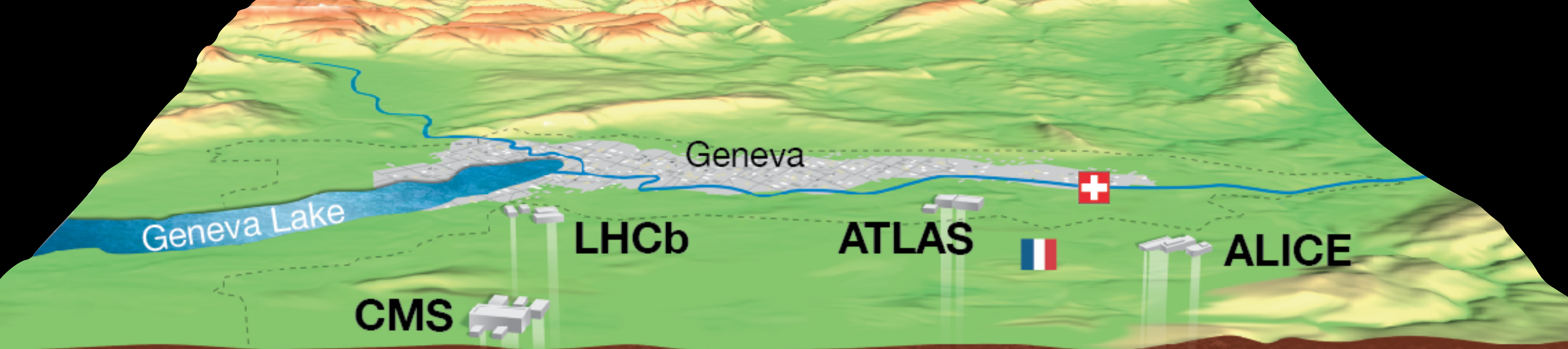
Data recorded: 2010-Nov-14 18:37:44.420271 GMT(19:37:44 CEST)

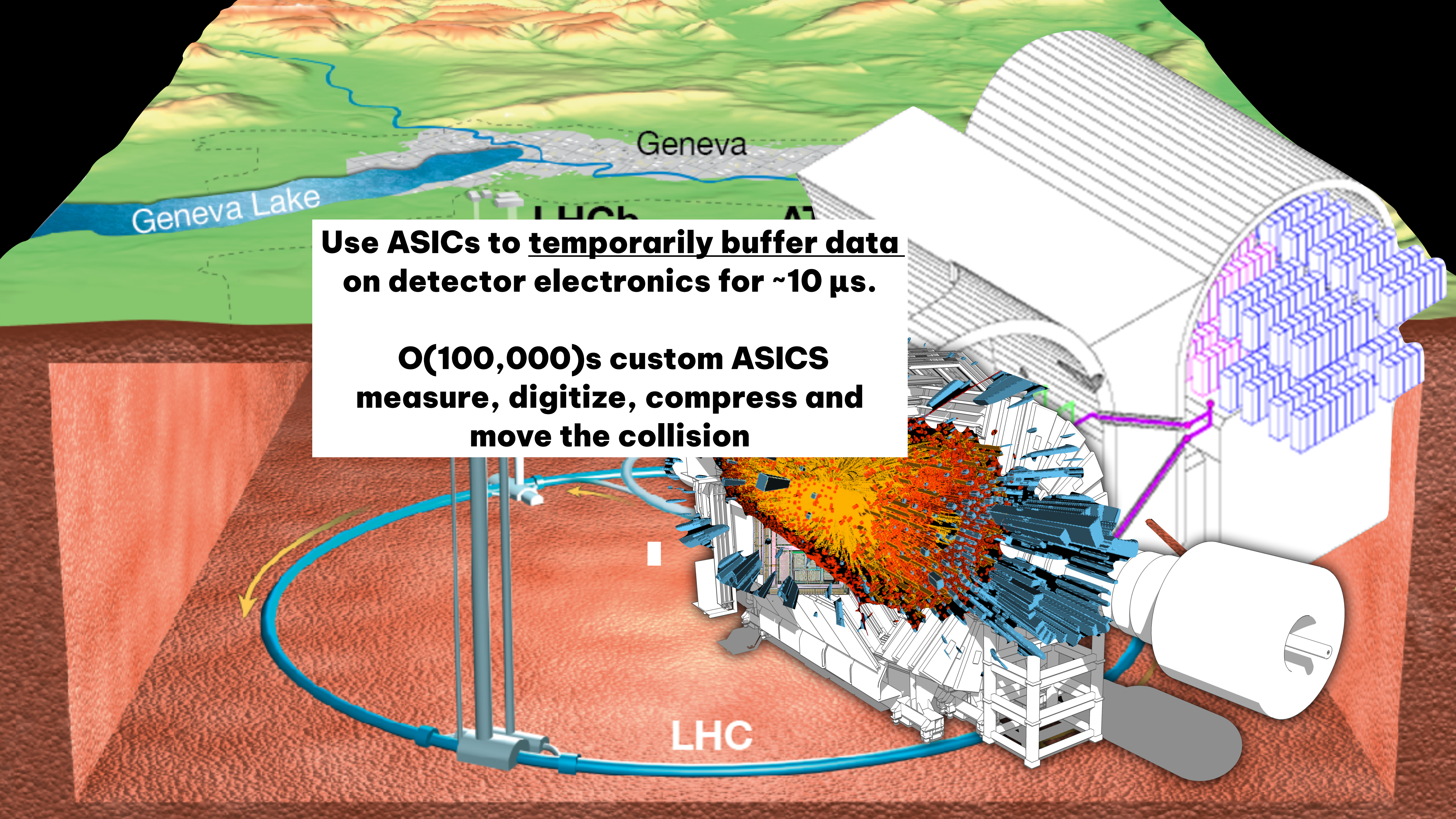
Run / Event: 151076 / 1405388

1 collision $\cong$ O(1) MB
 $\sim$ O(1) billion collisions per second
 $\rightarrow$ O(1) PB / second

– several times the average data traffic of the entire global Internet.
Before we even can think about doing science, need to think about this one







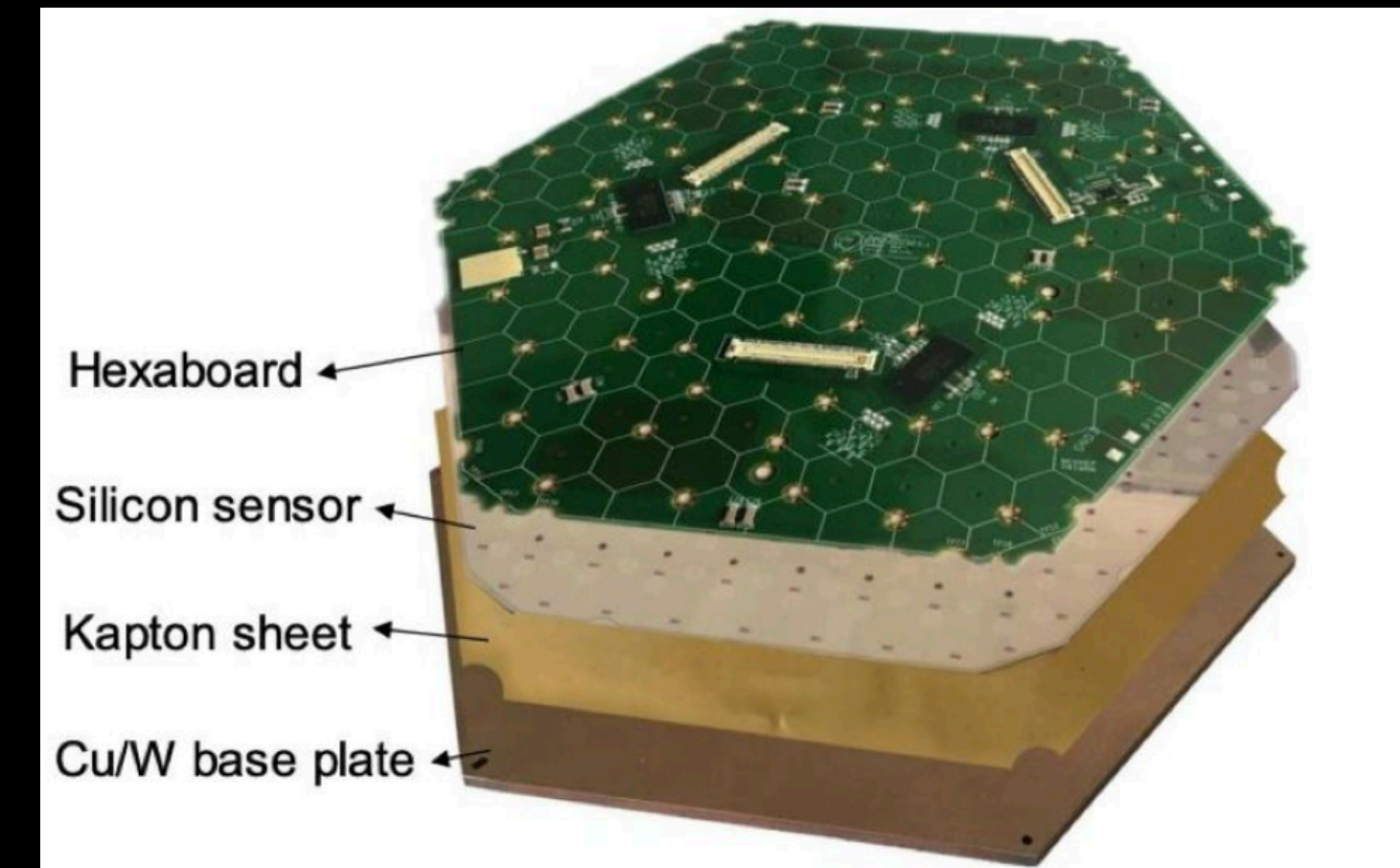
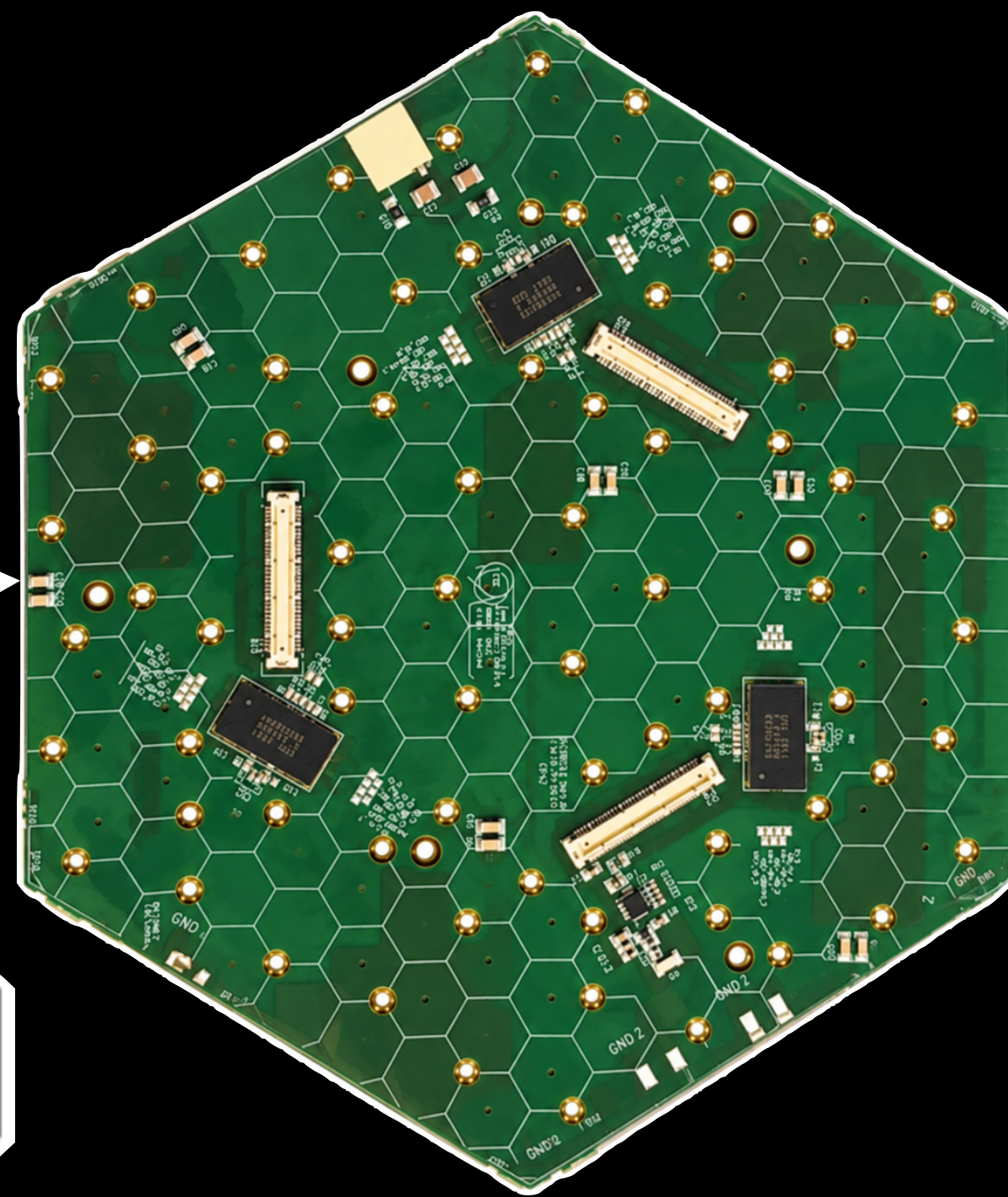
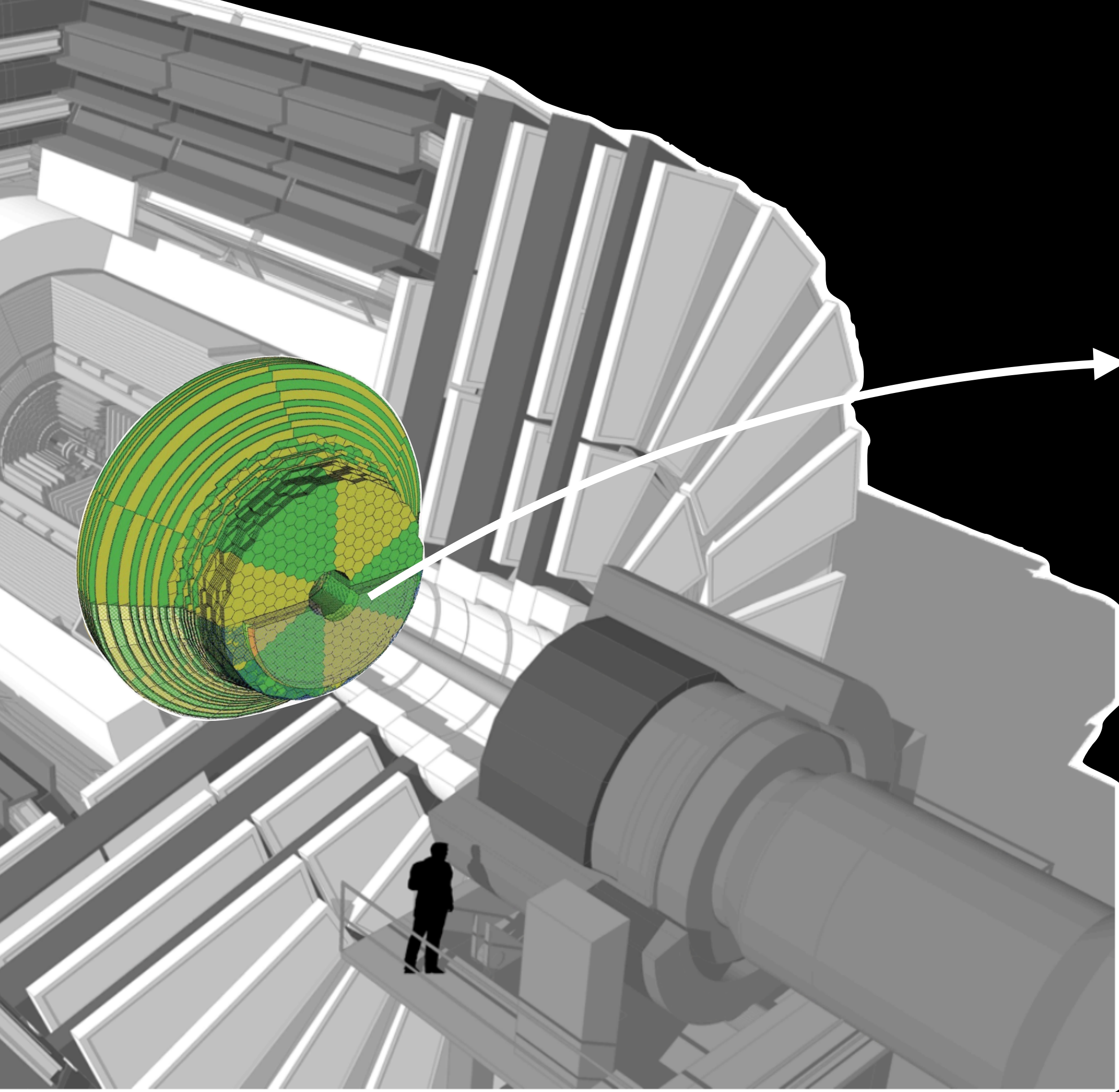
Geneva

Geneva Lake

**Use ASICs to temporarily buffer data
on detector electronics for $\sim 10 \mu\text{s}$.**

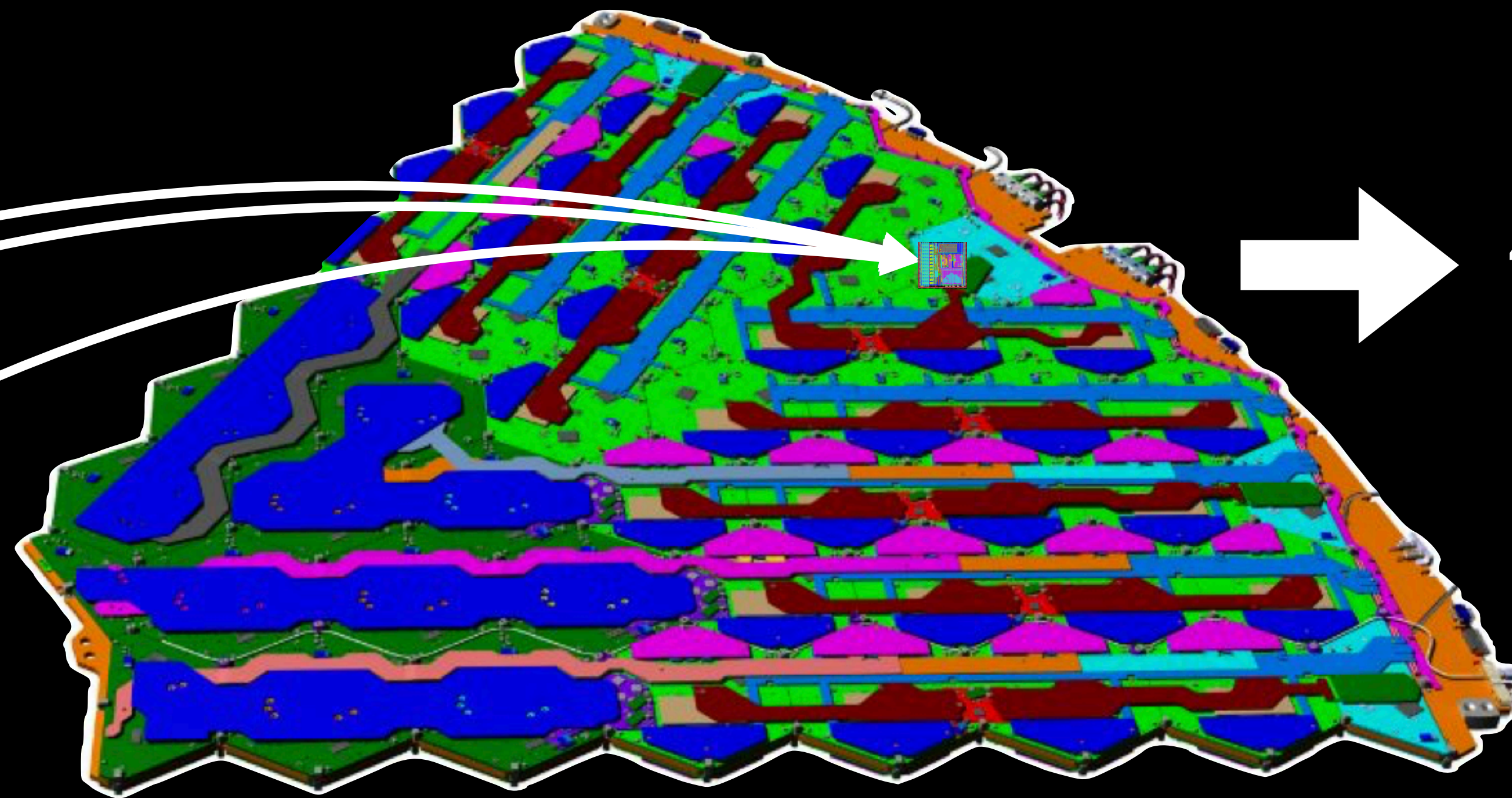
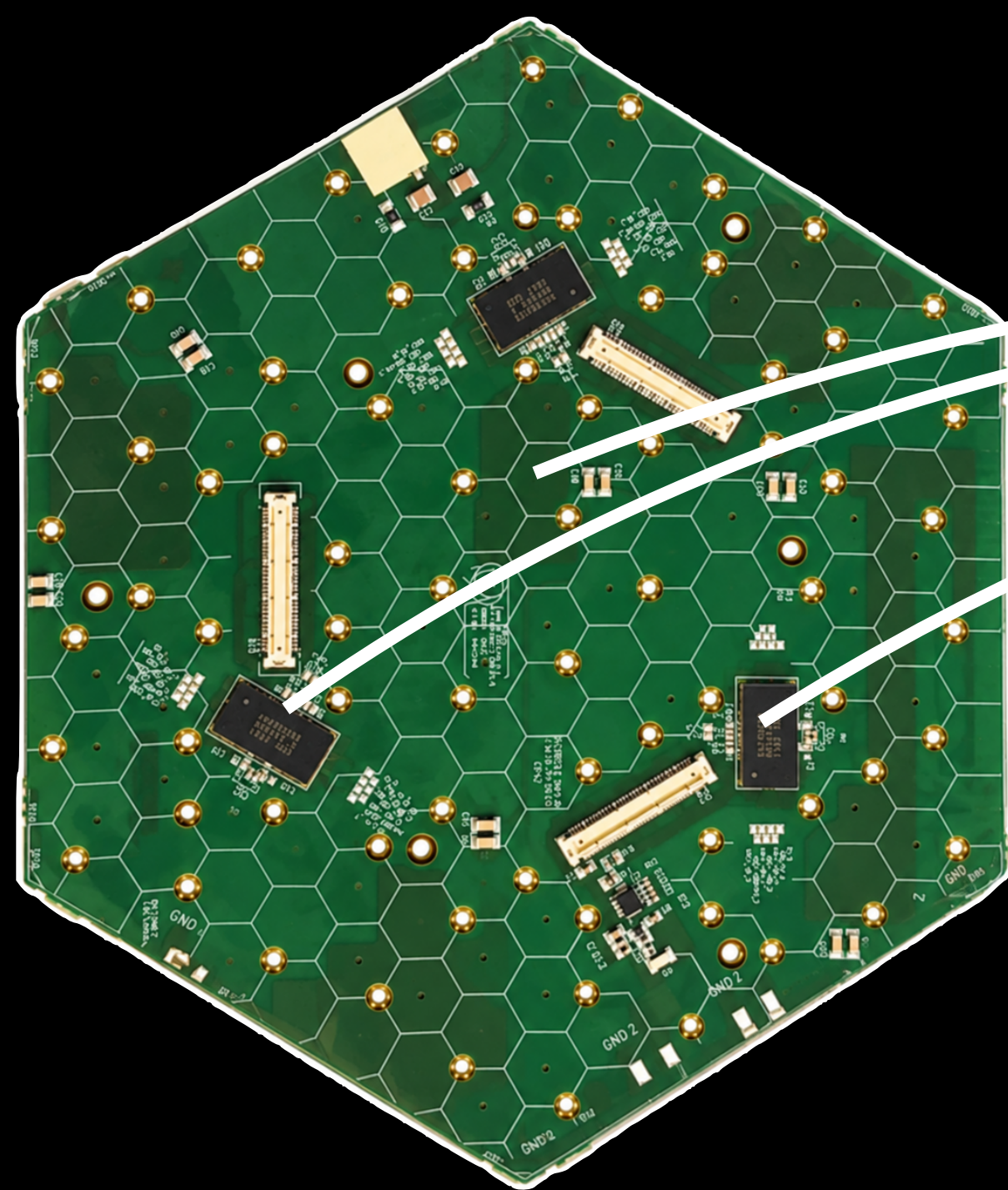
**$O(100,000)$ s custom ASICS
measure, digitize, compress and
move the collision**

LHC



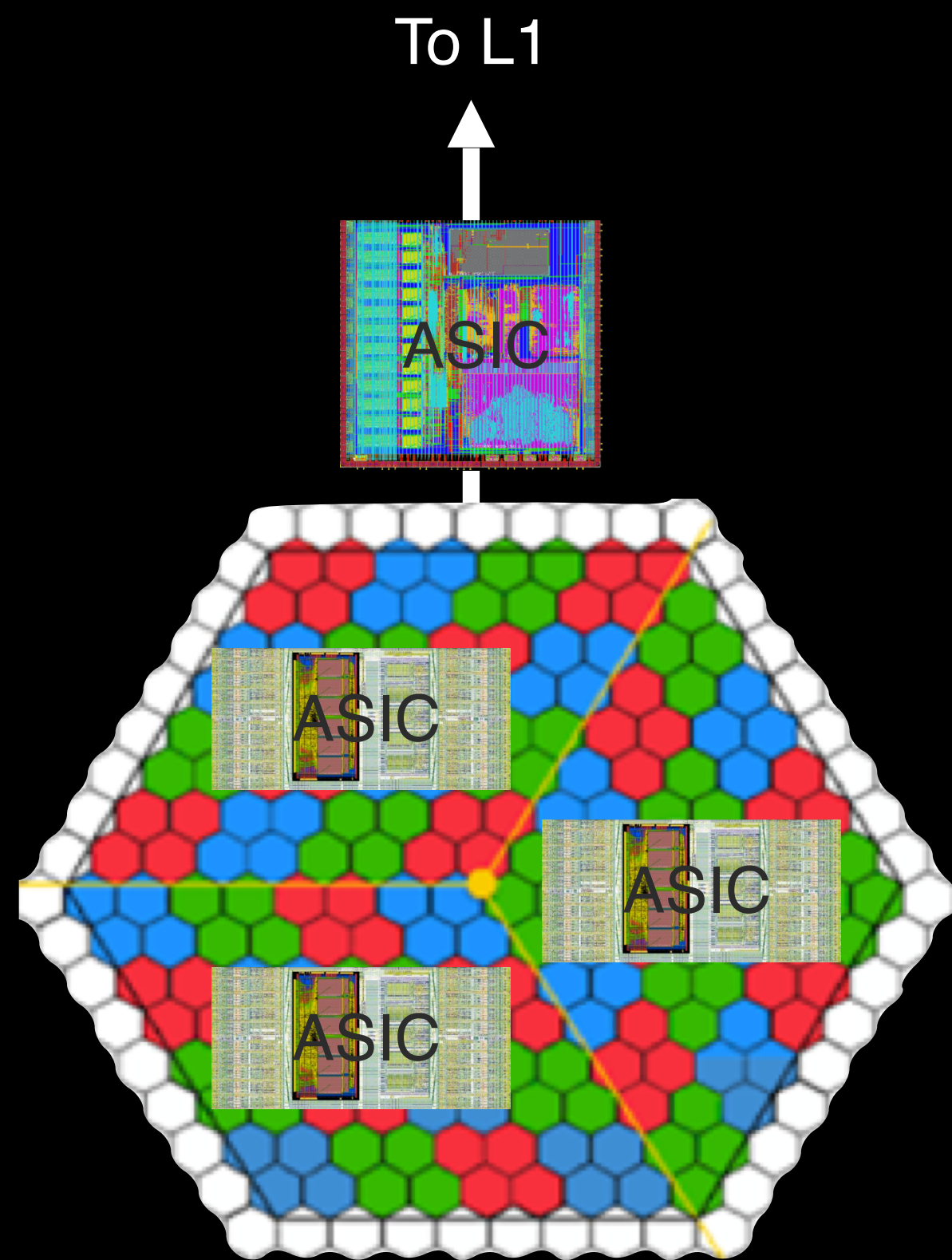
**10,000 sensor
modules per endcap.**
ASICs (HGCR0C)
glued onto sensor
sends cell charges out
@40 MHz

E.g. Forward calorimeter:
65 million channels spanning 600 m²



45 Tb/s to event
filter system

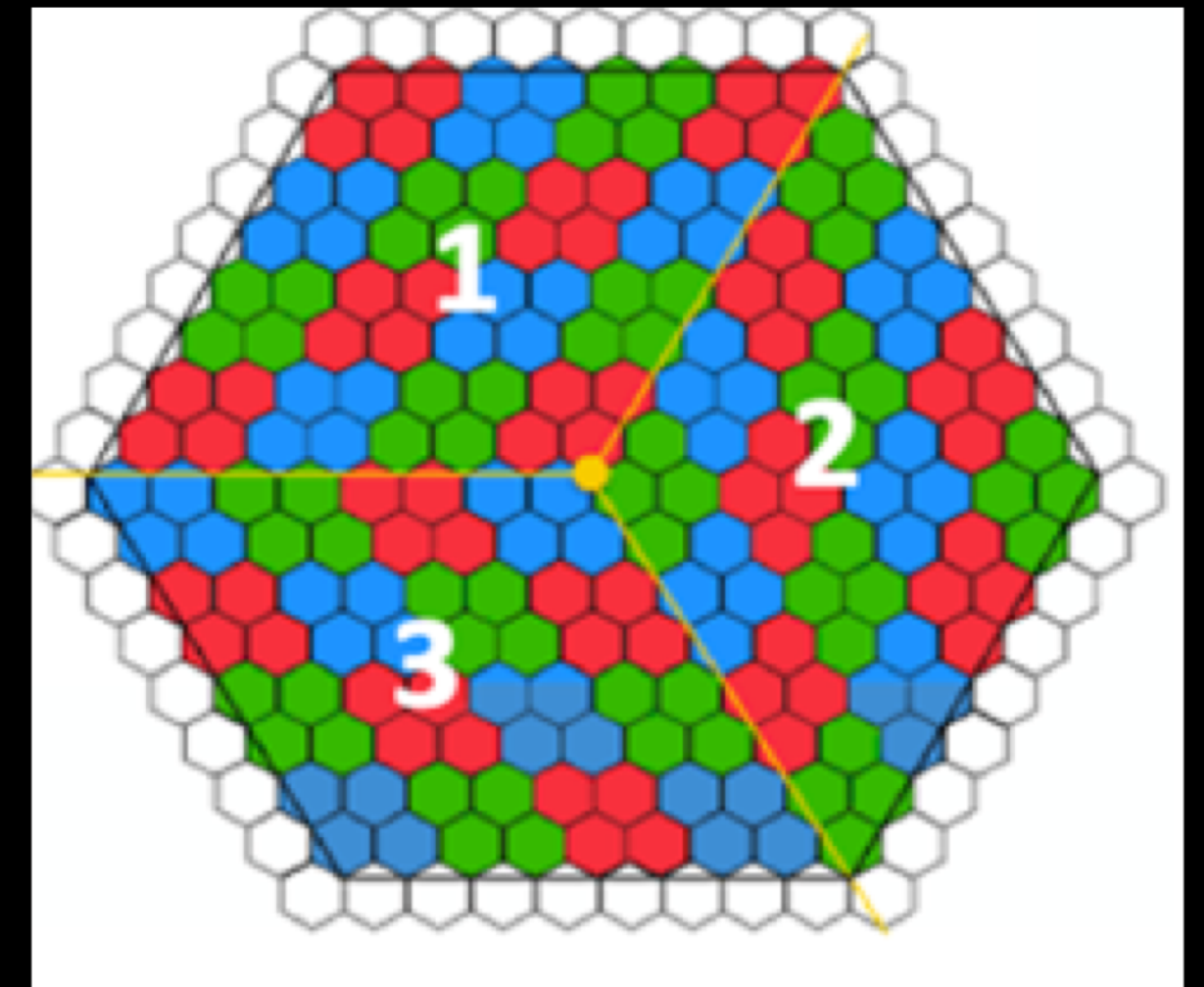
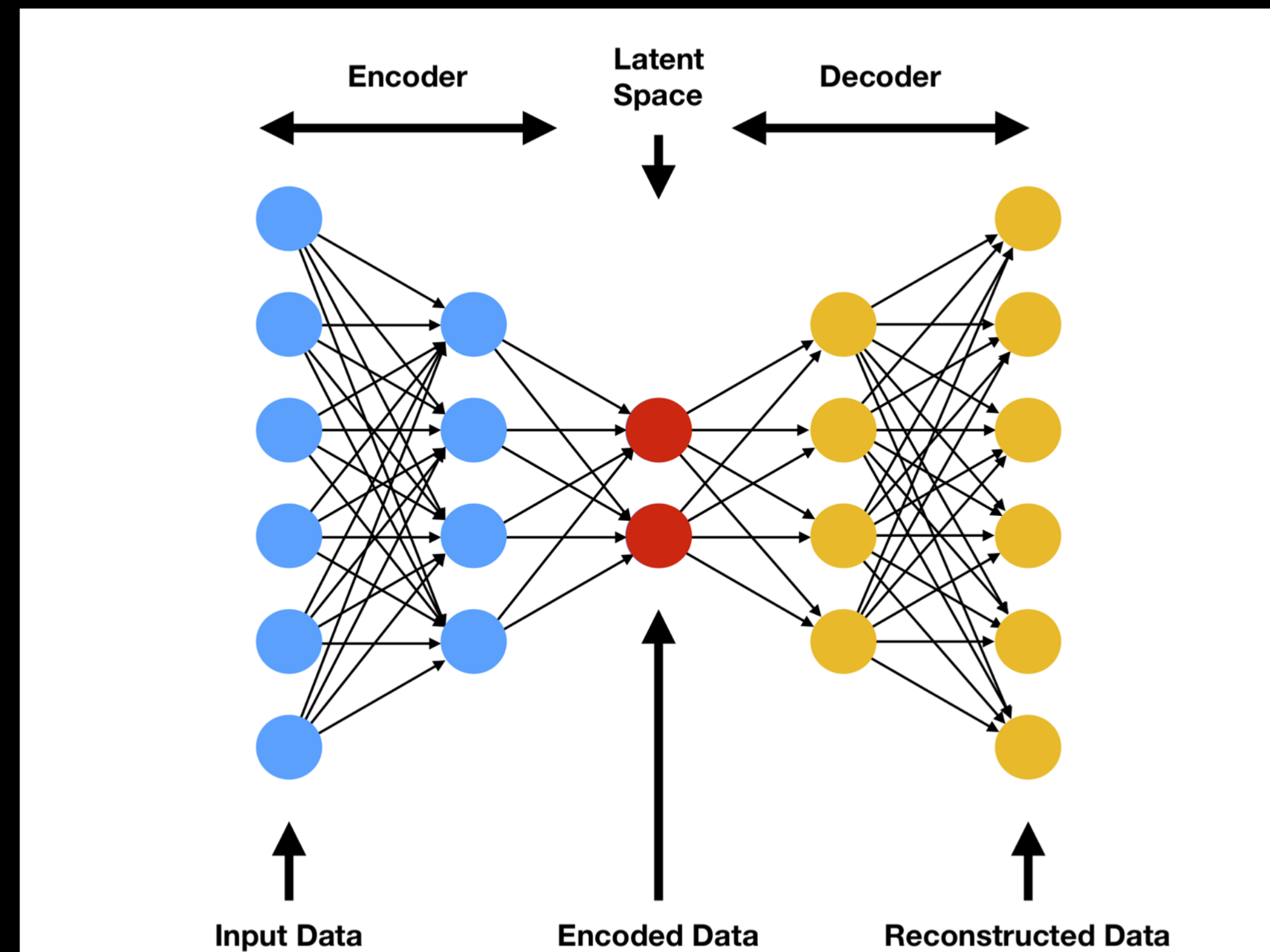
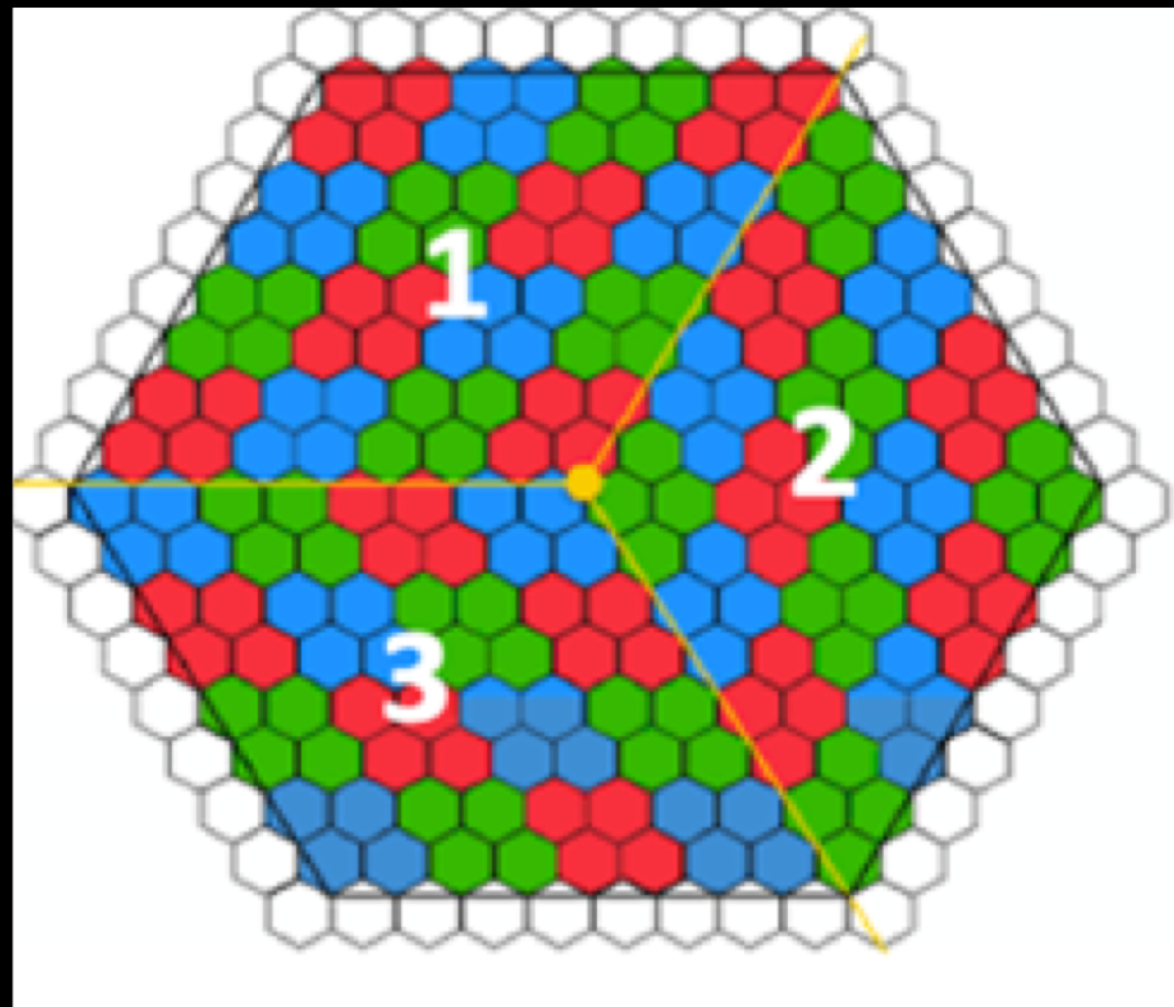
32-bit “word” from each HGCROC to
concentrator ASIC (ECON-T)
**must compress this data in <400 ns in
order to move it out of the detector**



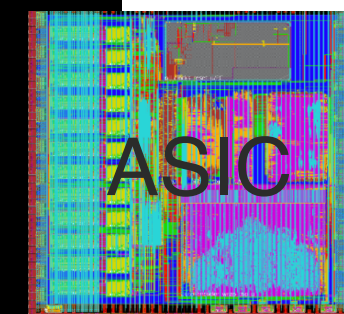
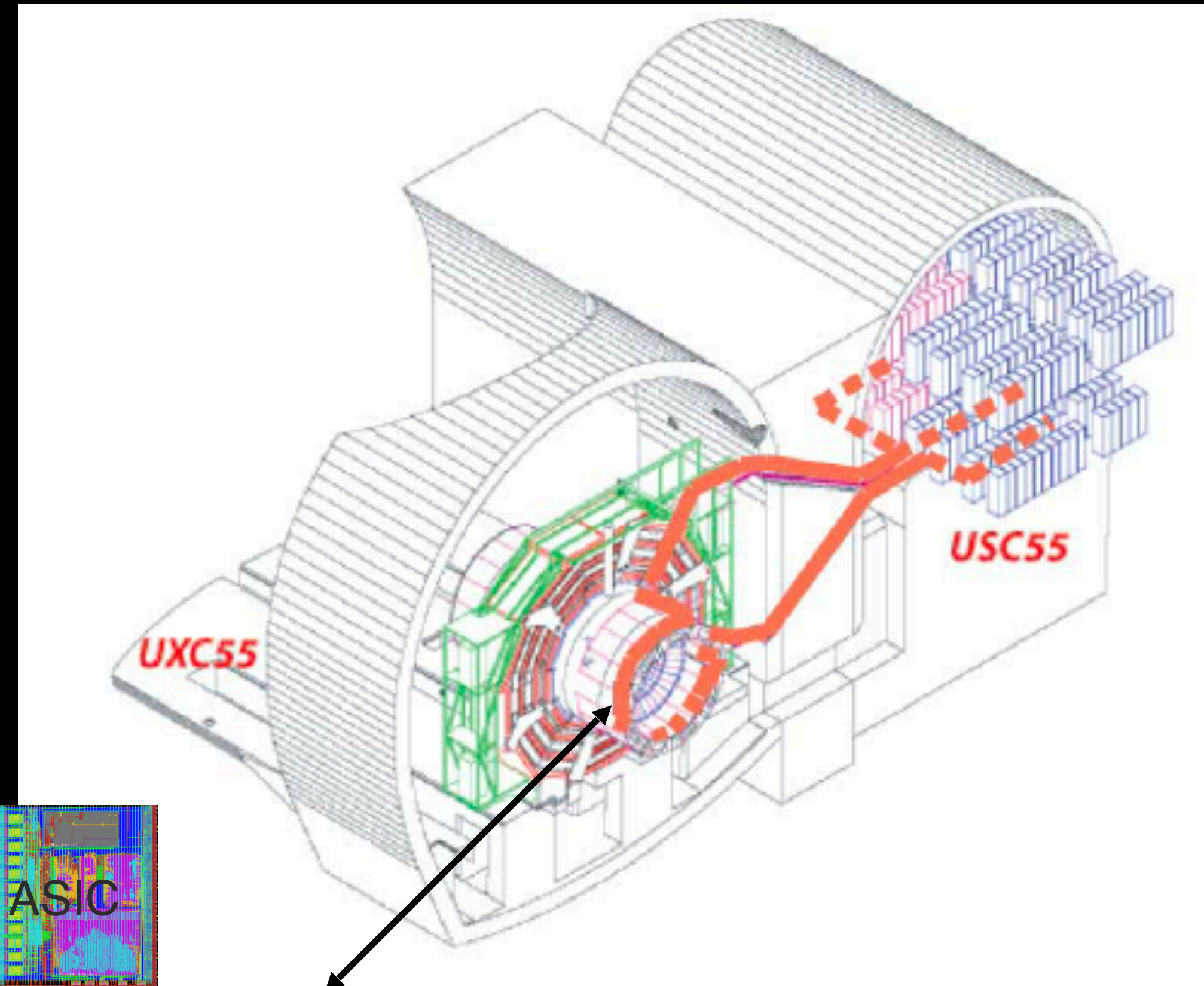
Must compress ON DETECTOR

- Extreme radiation (660 Mrad)
- Cooled to -30 degrees
- 400 ns latency budget

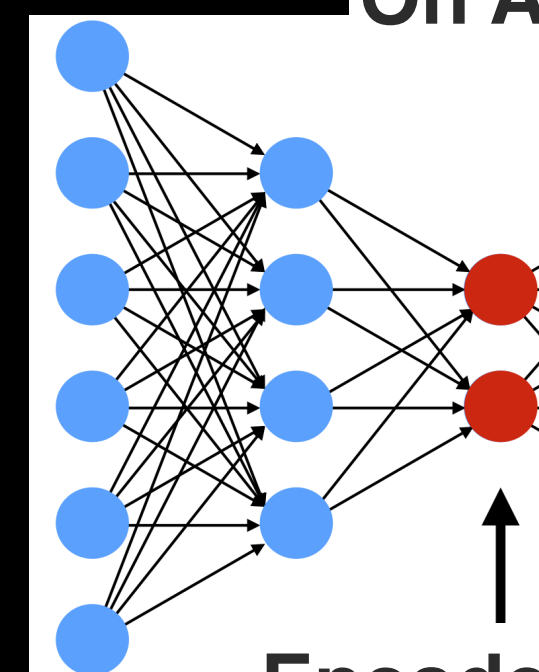
Nanosecond ML for compression on ASIC



Variational Autoencoder



On ASIC

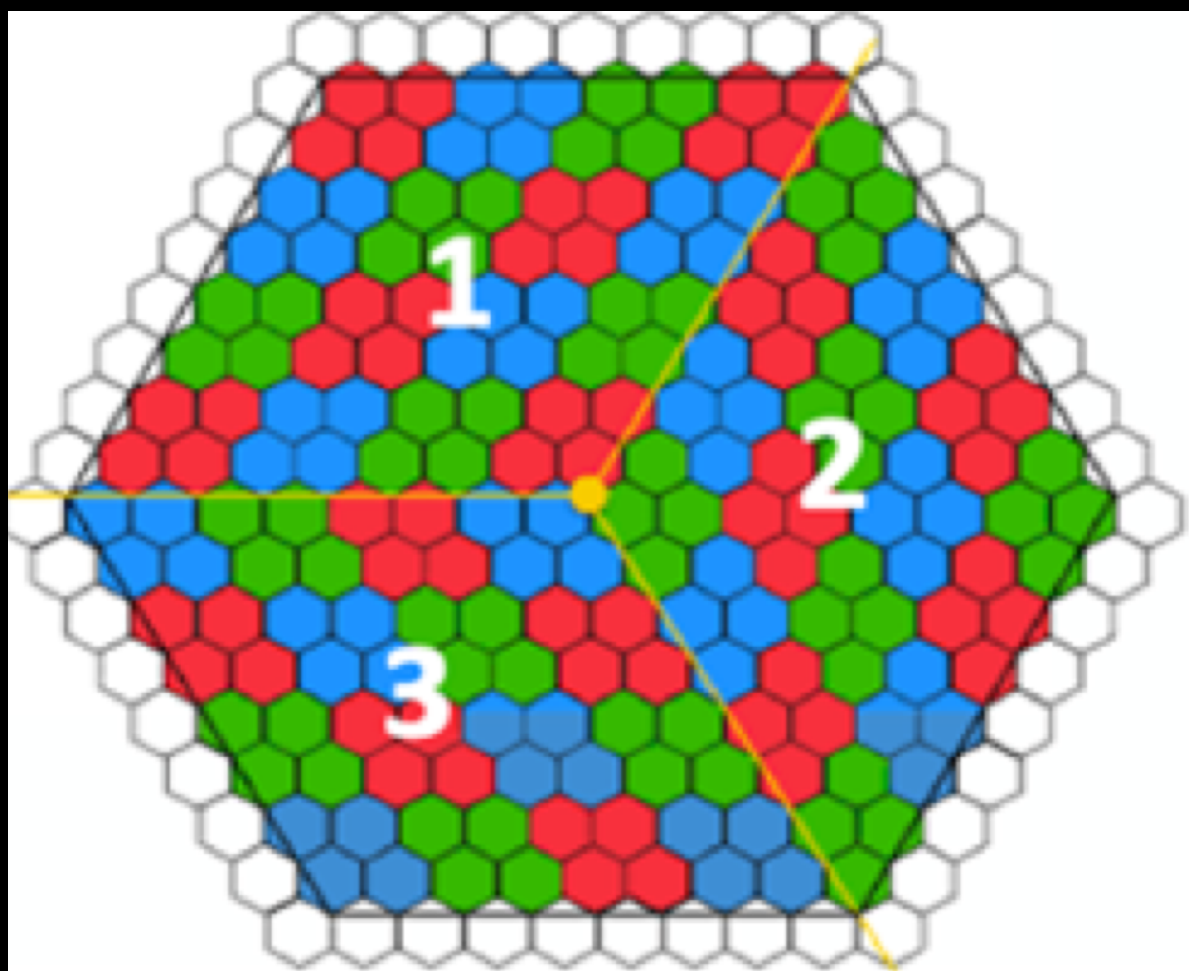


Encoded data

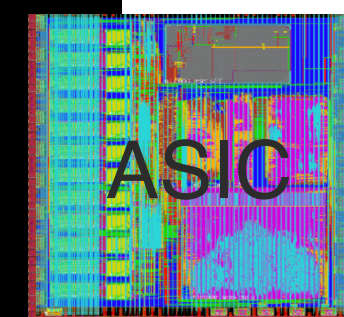
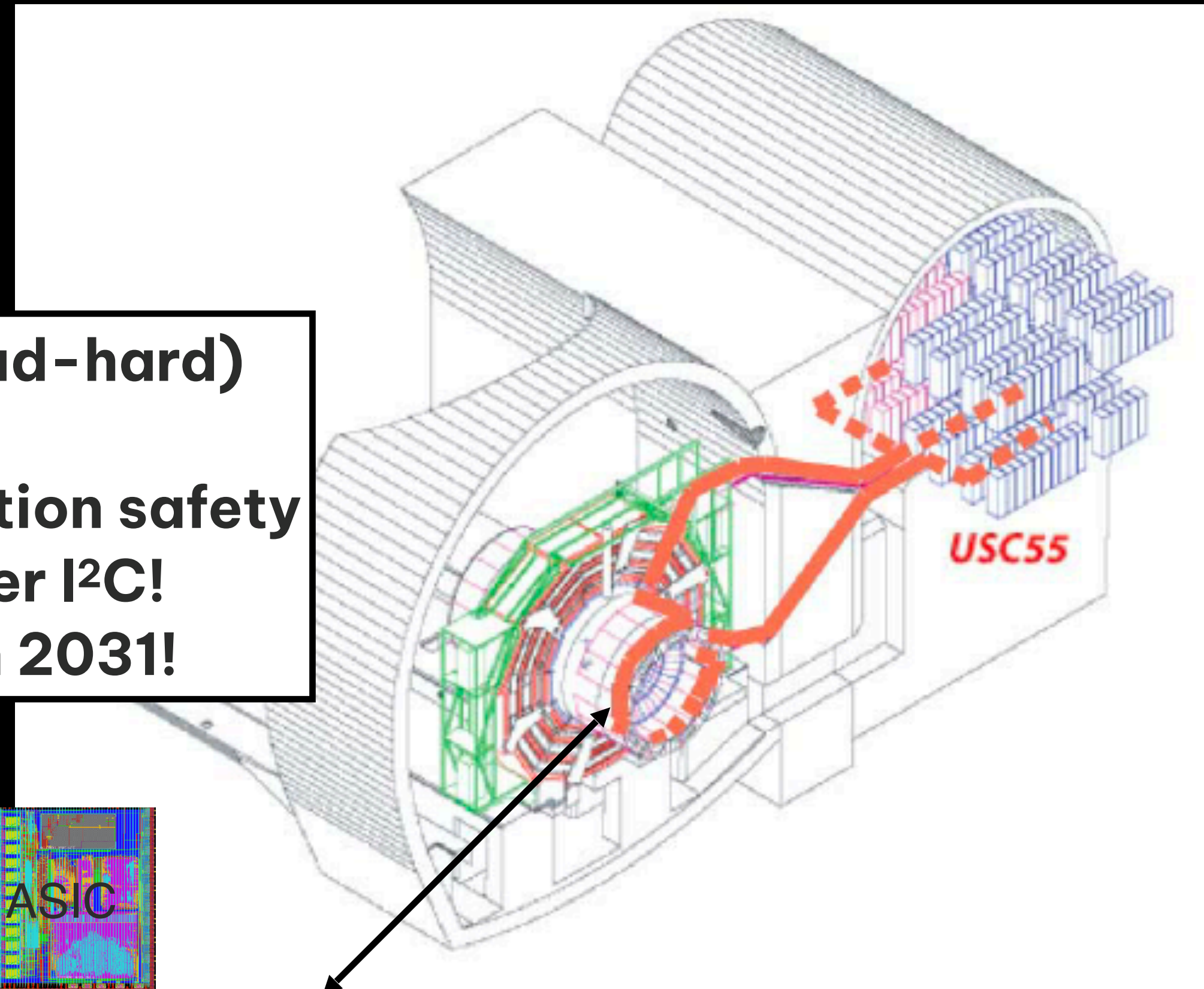
Transmit encoded data!
 $O(10)$ Tb/s at 40 MHz



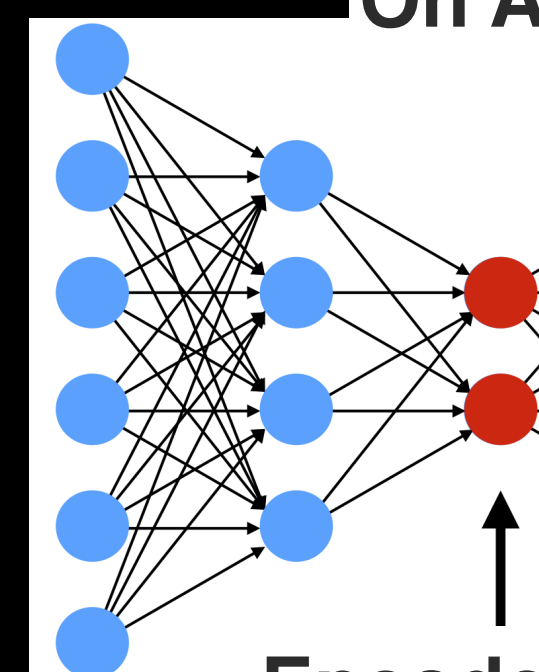
(Will talk about the tools
we made to do this later)



- ASIC inference (65 nm, rad-hard)
- < 2.5 mW/channel
- Triplicated w/b for radiation safety
Reprogrammable w/b over I²C!
- 20,000 going into CMS in 2031!

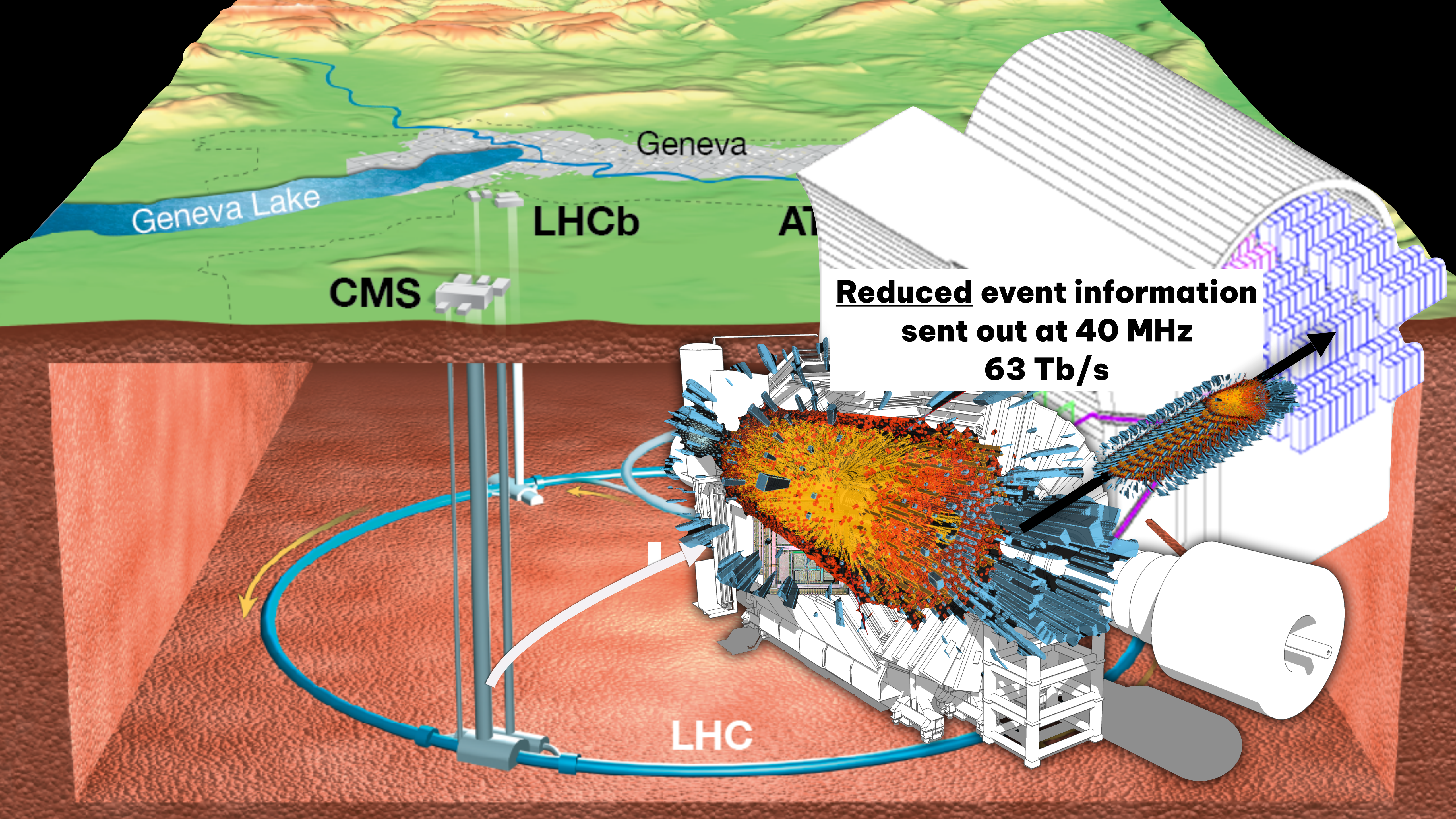


On ASIC



Transmit encoded data!
 $O(10)$ Tb/s at 40 MHz

(Will talk about the tools
we made to do this later)



Geneva

Geneva Lake

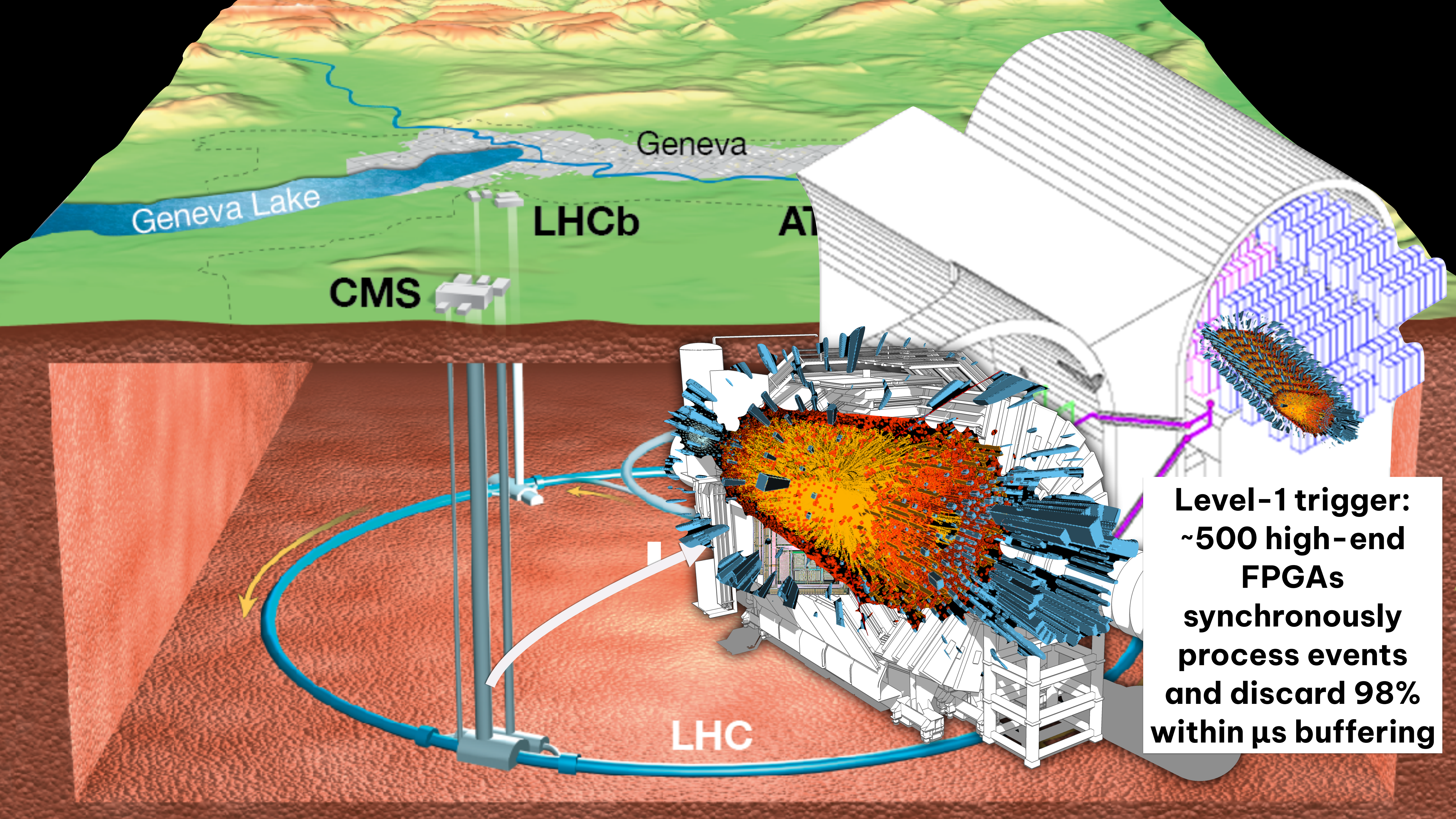
LHCb

ATLAS

CMS

**Reduced event information
sent out at 40 MHz
63 Tb/s**

LHC



Geneva Lake

Geneva

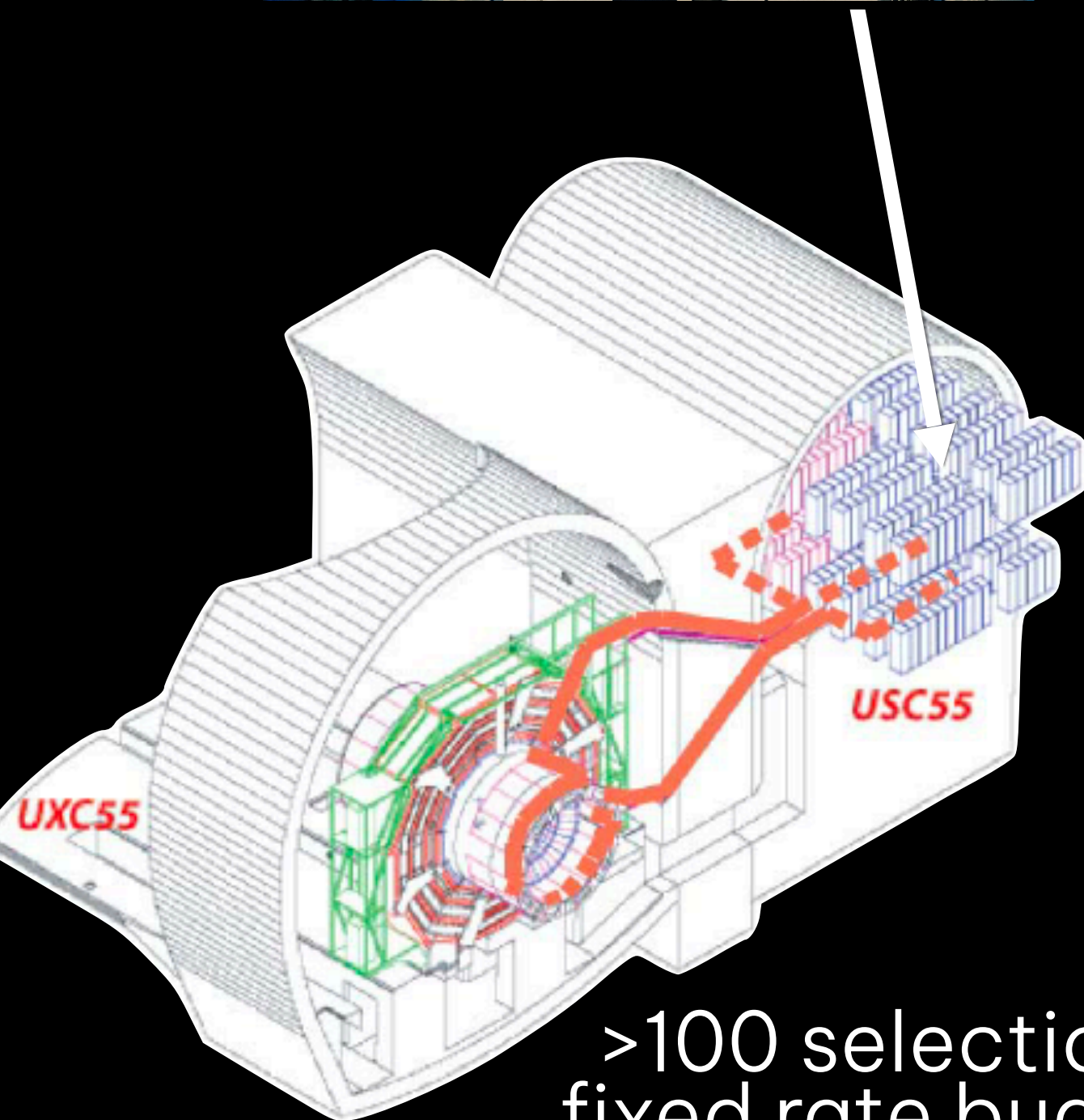
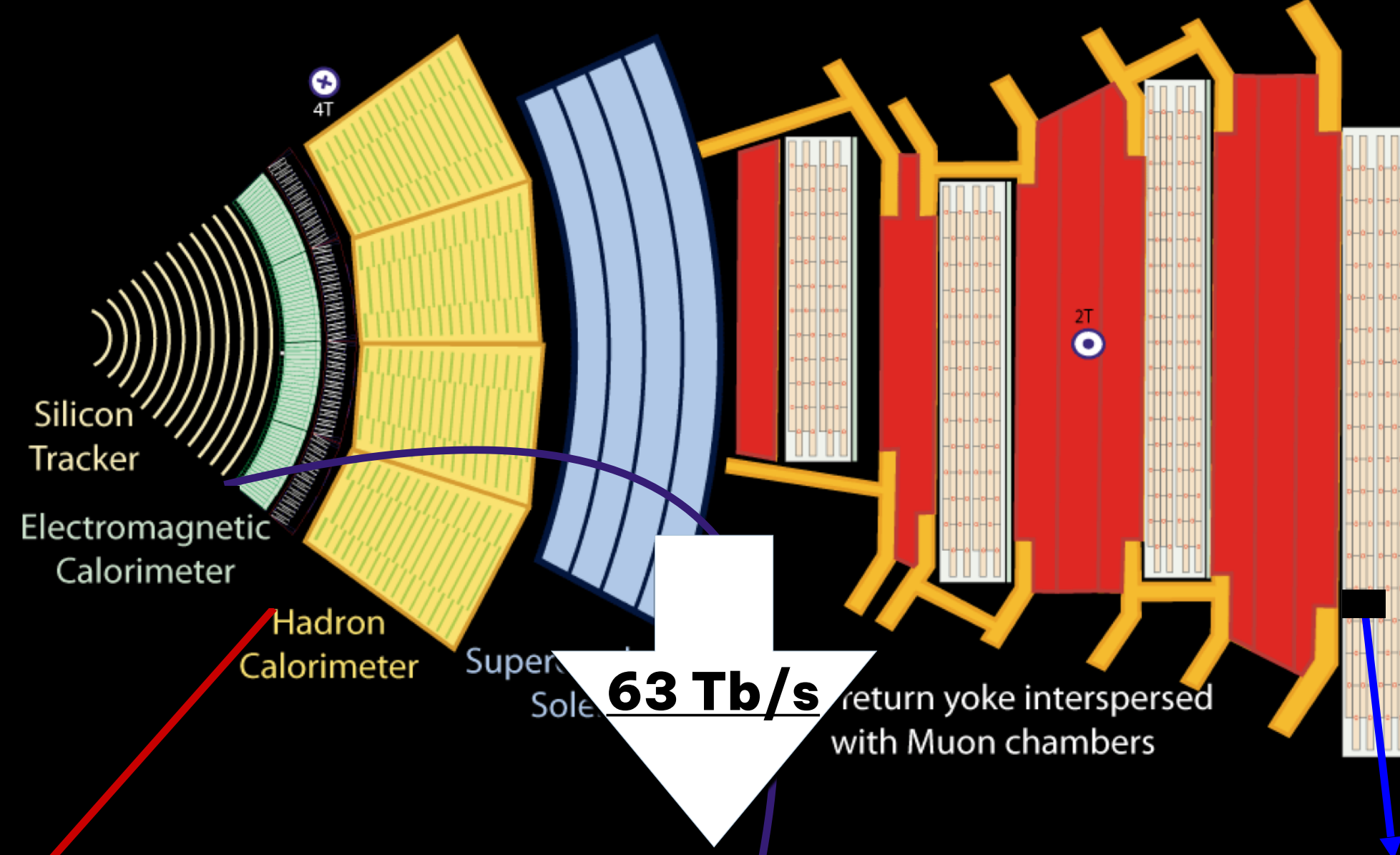
LHCb

ATLAS

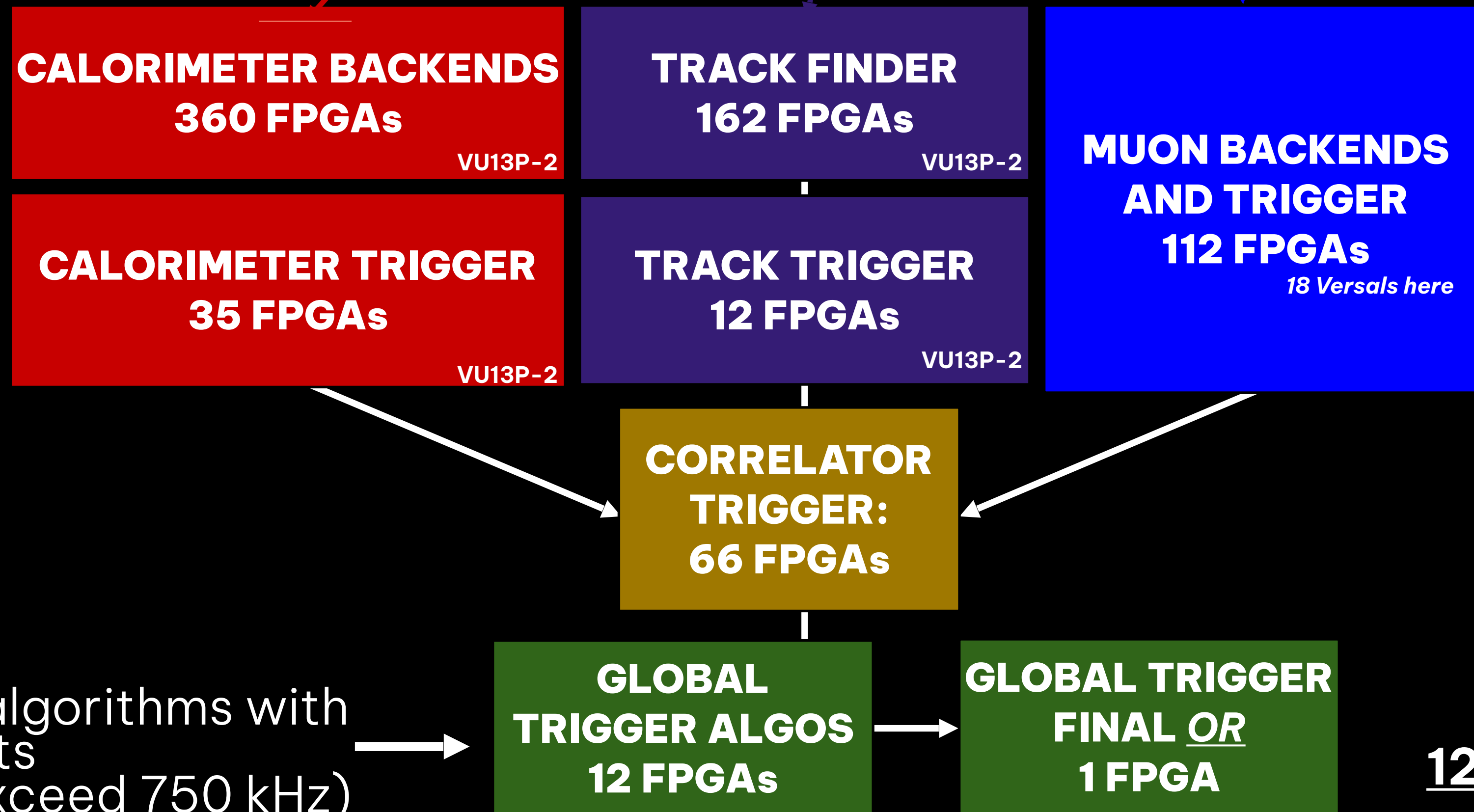
CMS

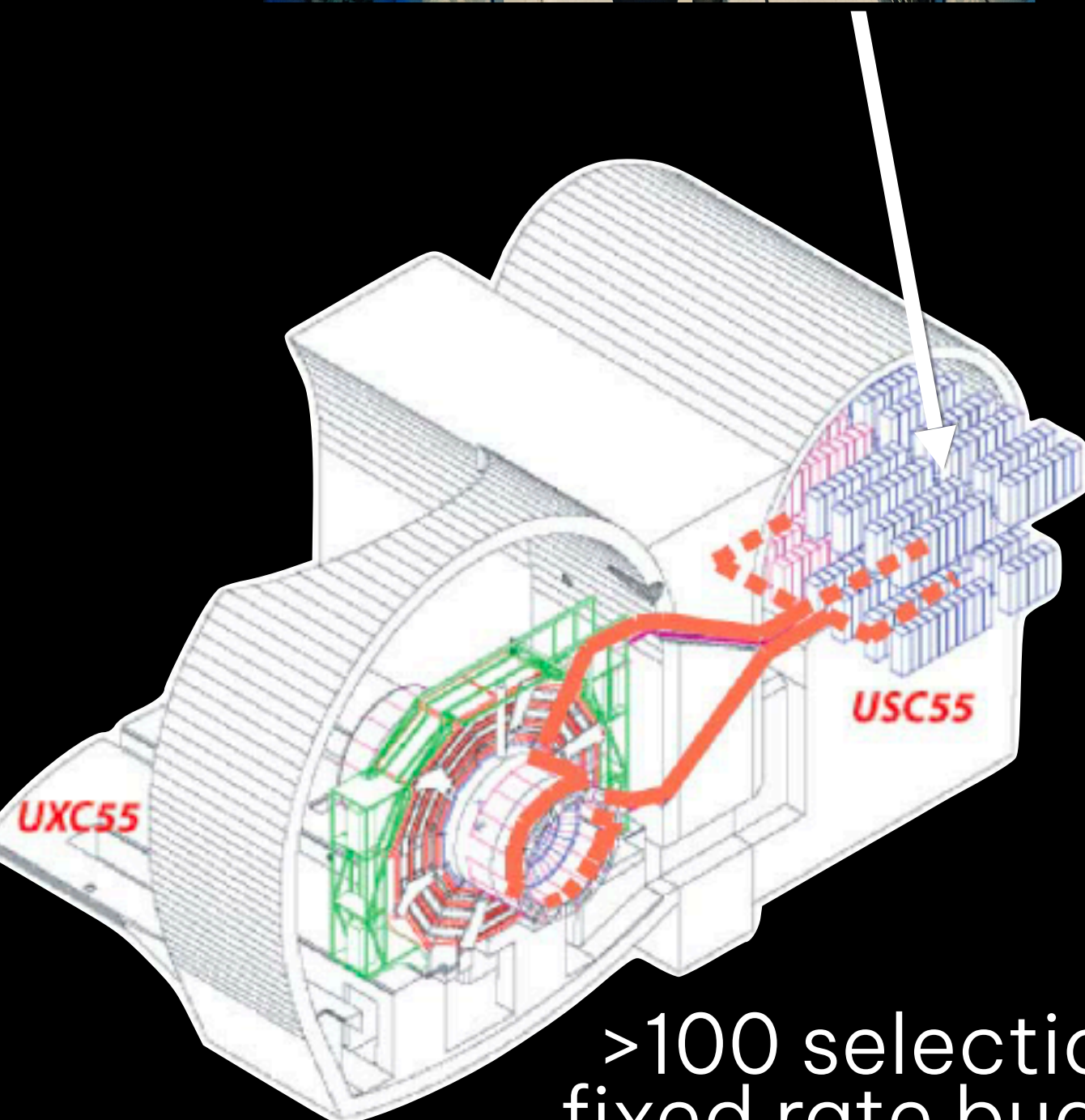
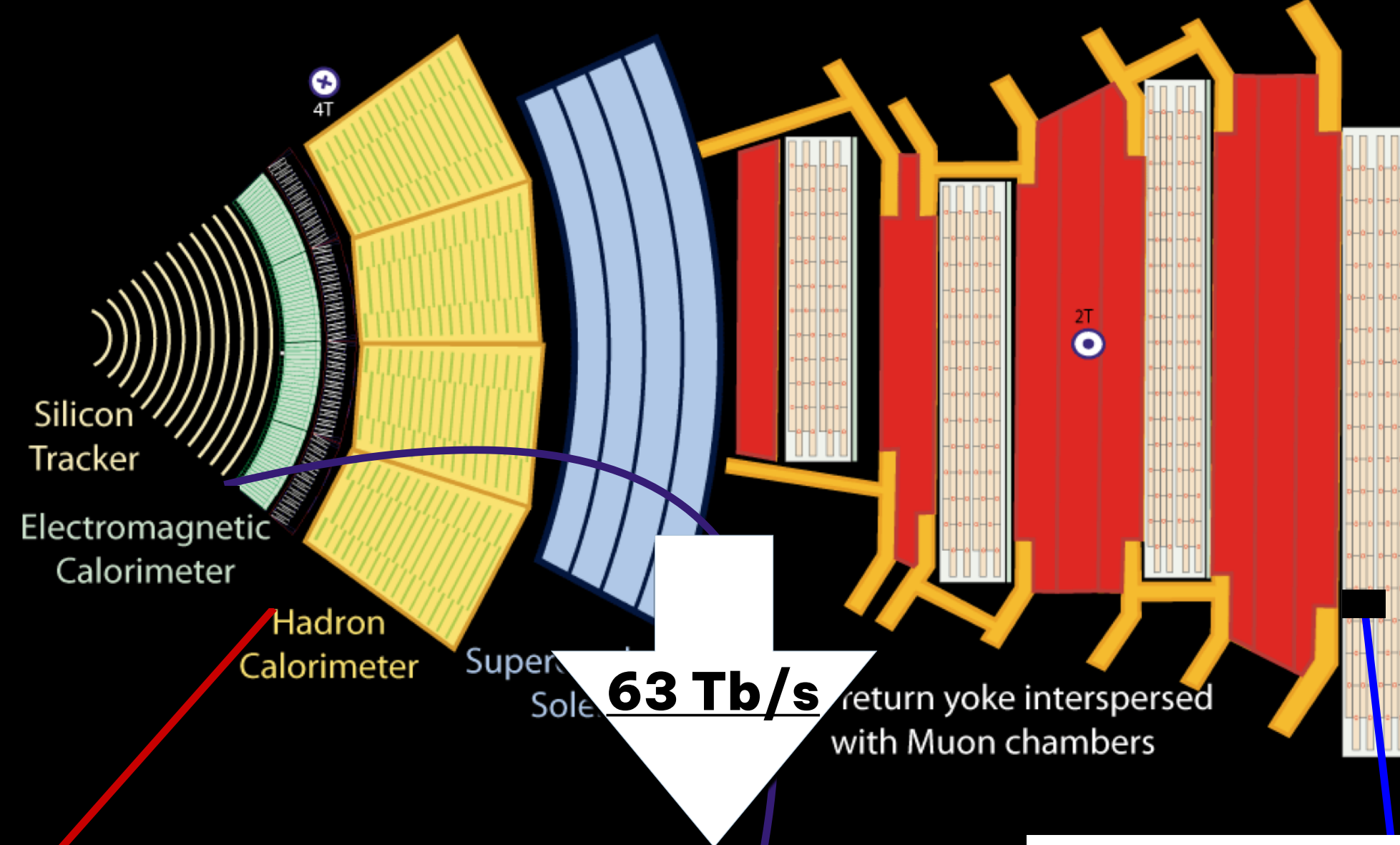
LHC

Level-1 trigger:
~500 high-end
FPGAs
synchronously
process events
and discard 98%
within μs buffering



>100 selection algorithms with fixed rate budgets
(total cannot exceed 750 kHz)





>100 selection algorithms with fixed rate budgets (total cannot exceed 750 kHz)

CALORIMETER BACKENDS
360 FPGAs

VU13P-2

CALORIMETER TRIGGER
35 FPGAs

VU13P-2

TRACK FINDER
162 FPGAs

VU13P-2

TRACK TRIGGER
12 FPGAs

VU13P-2

Lower-level reconstruction tasks

**MUON BACKENDS
AND TRIGGER**
112 FPGAs

18 Versals here

**CORRELATOR
TRIGGER:**
66 FPGAs

Higher-level decision making tasks

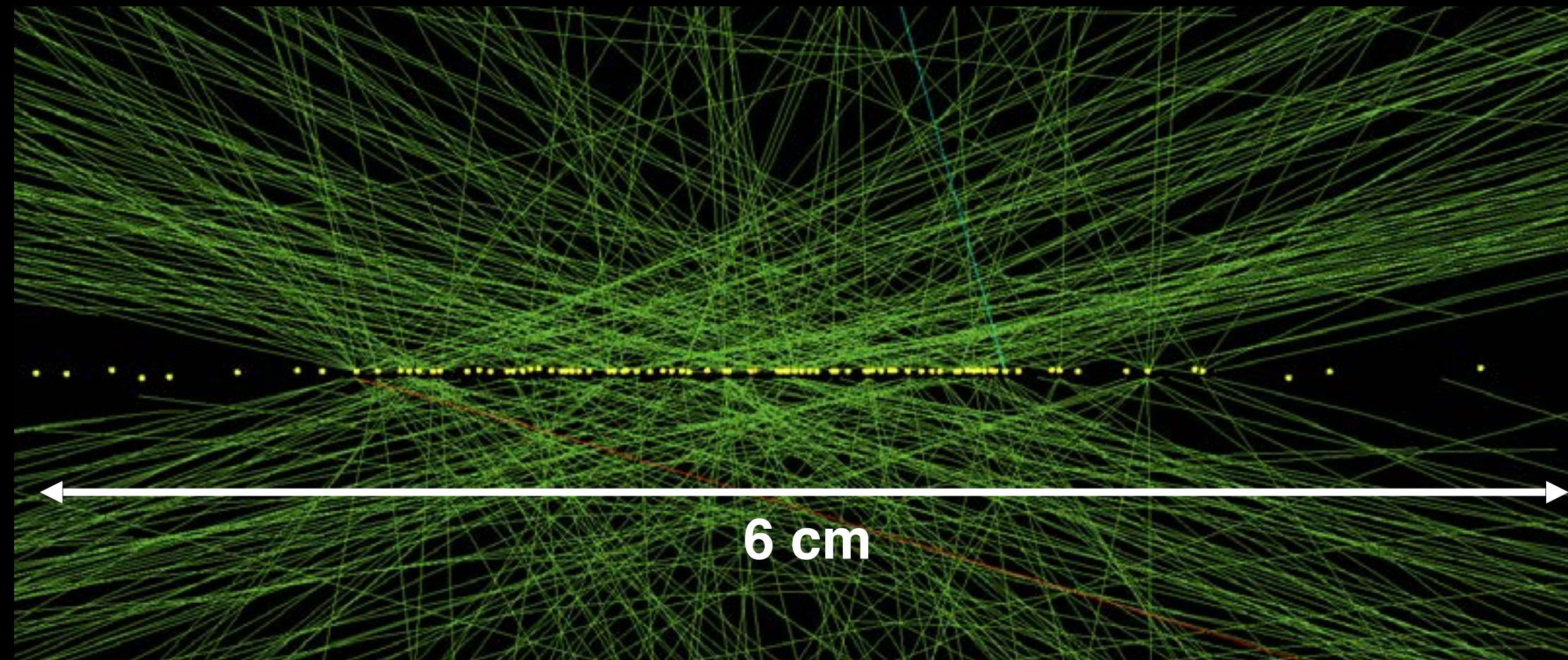
**GLOBAL
TRIGGER ALGOS**
12 FPGAs

**GLOBAL TRIGGER
FINAL OR**
1 FPGA

12.5 μ s

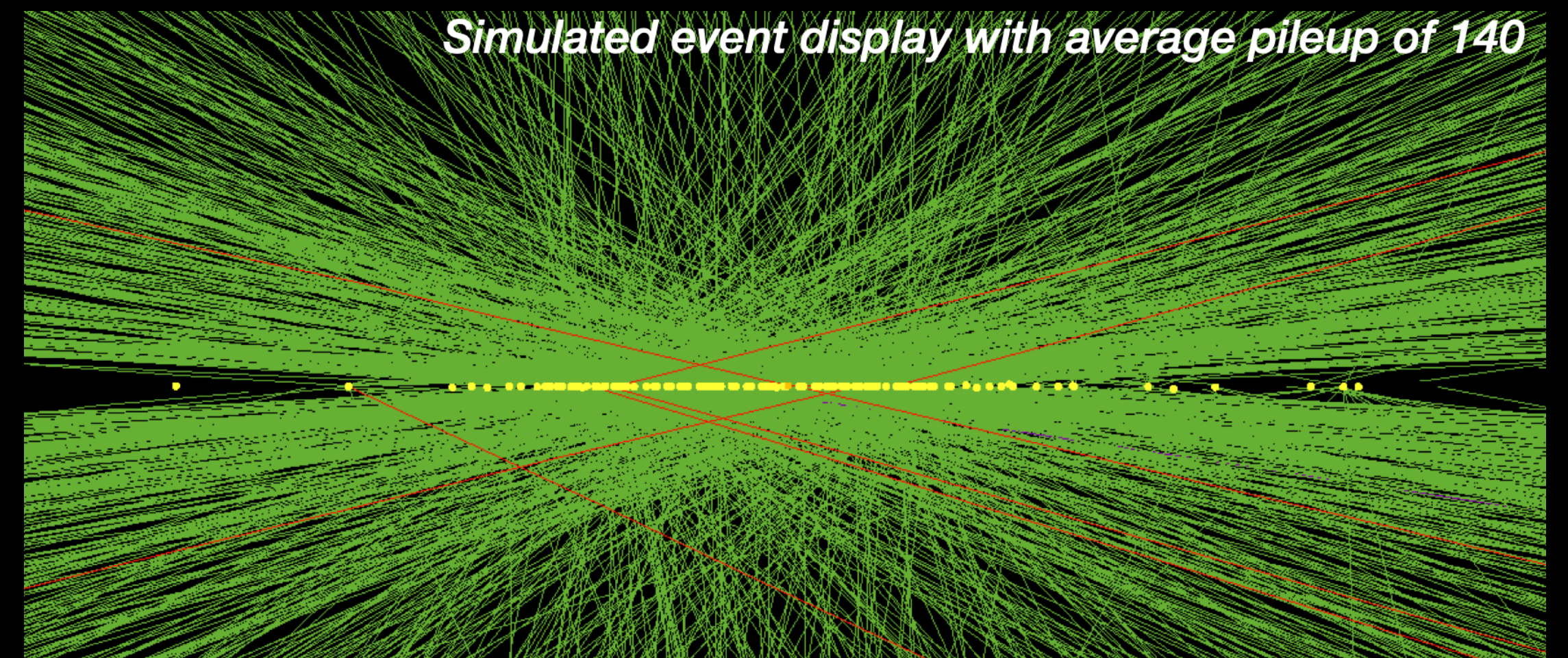
LHC (currently)

78 vertices
(average 60)



High Luminosity LHC (2030-2041)

200 vertices
(average 140)



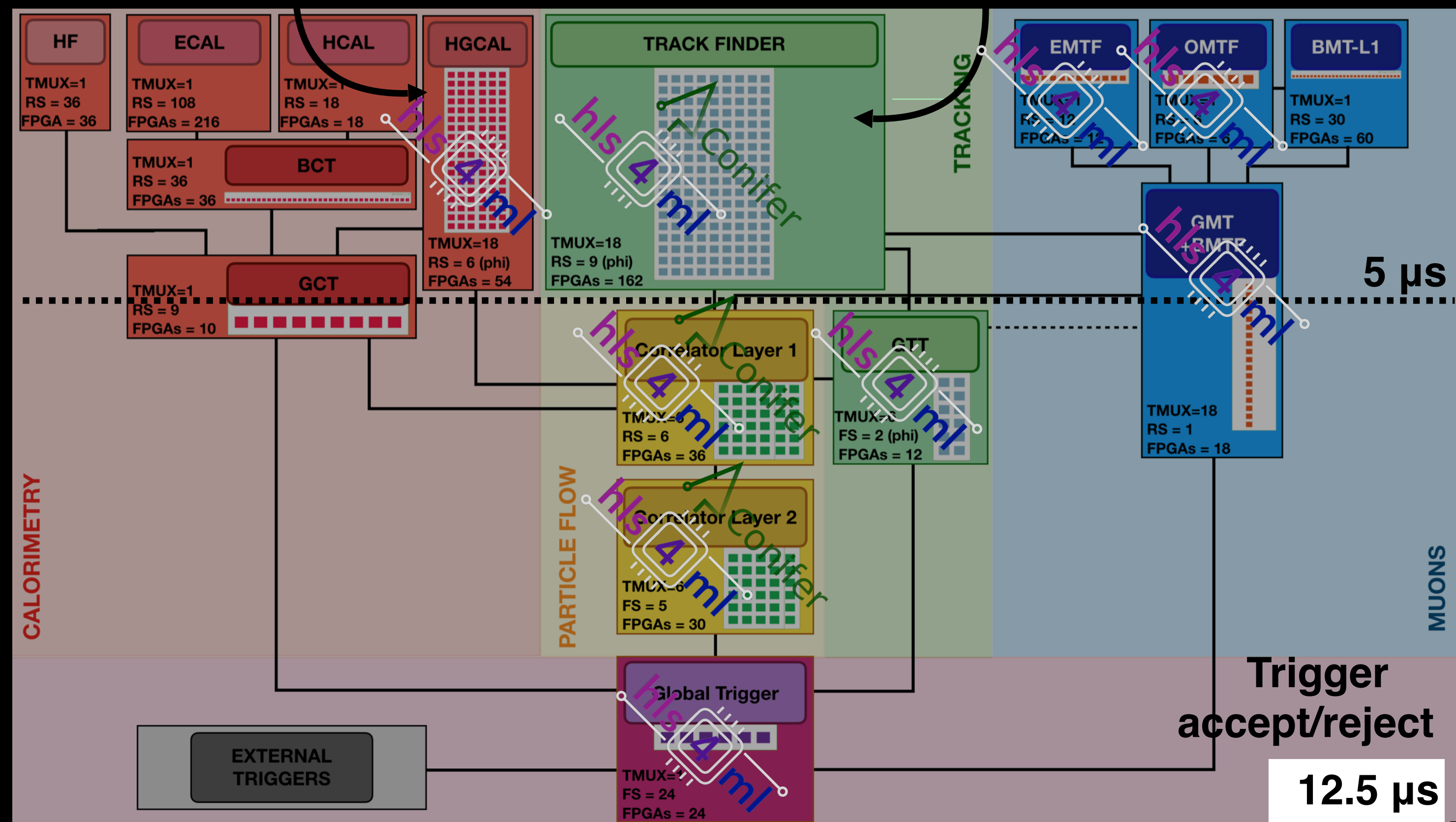
This is really really difficult!

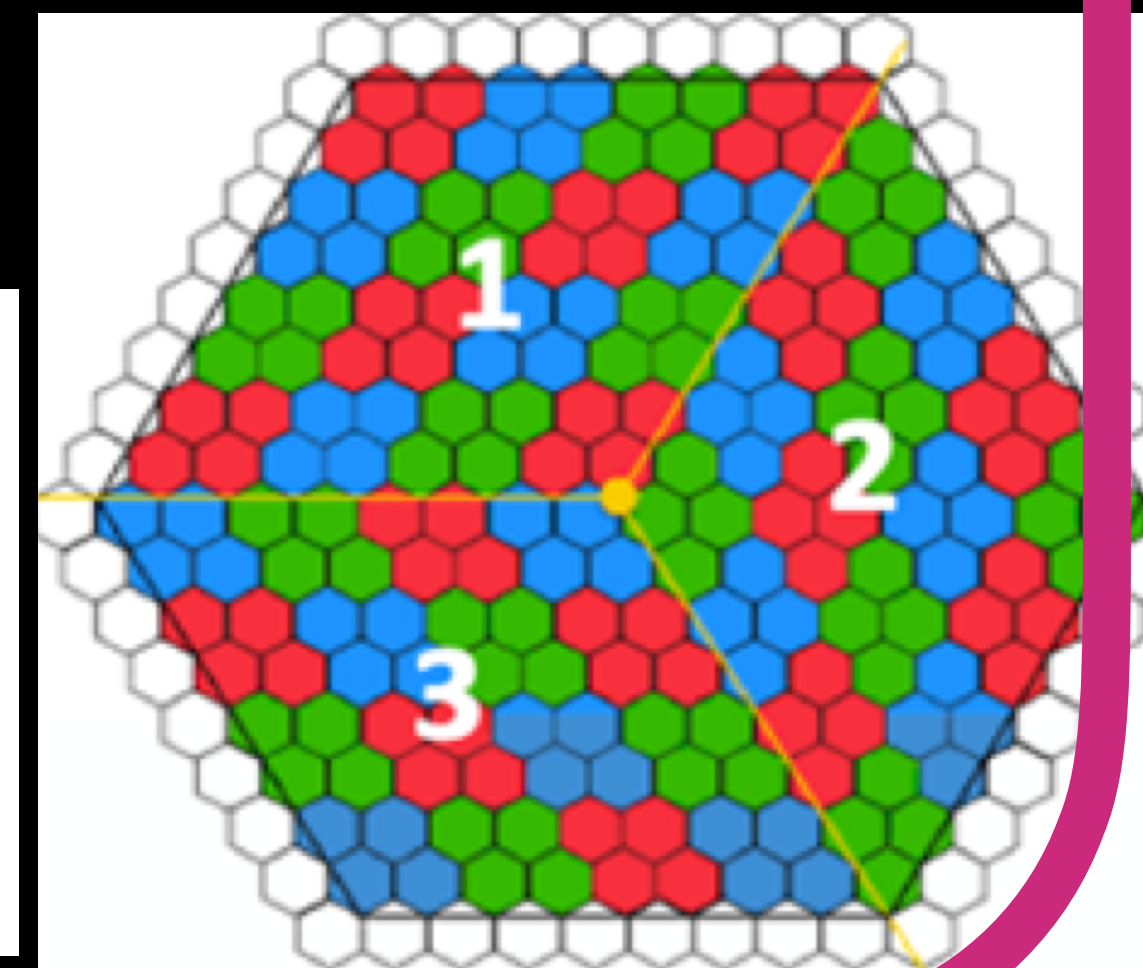
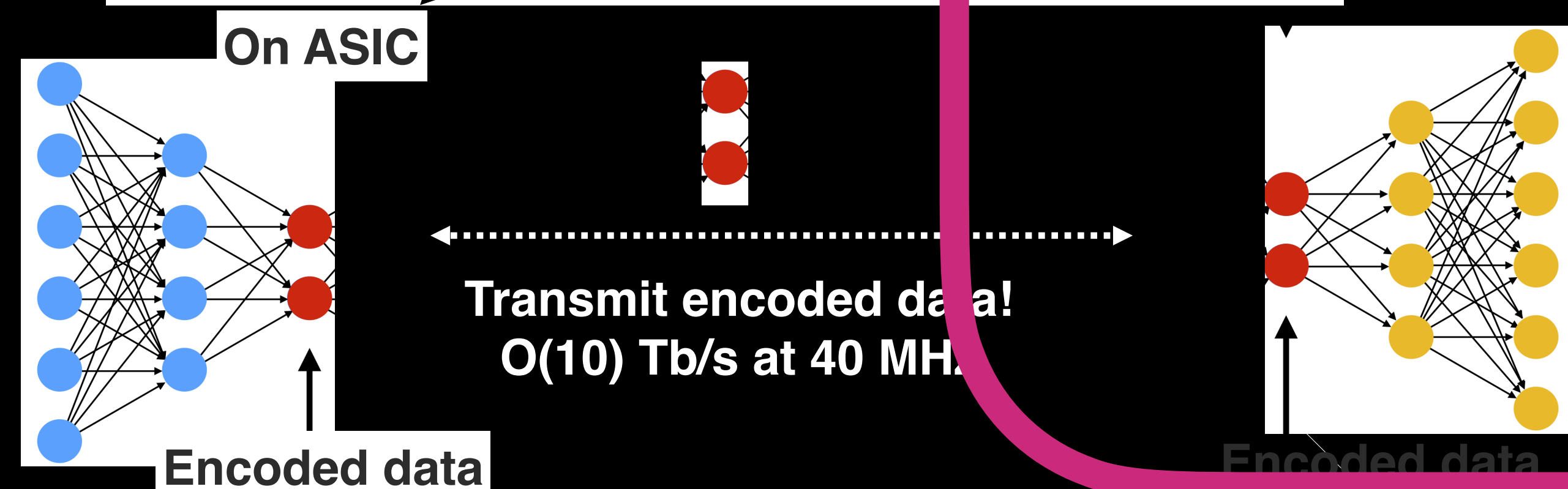
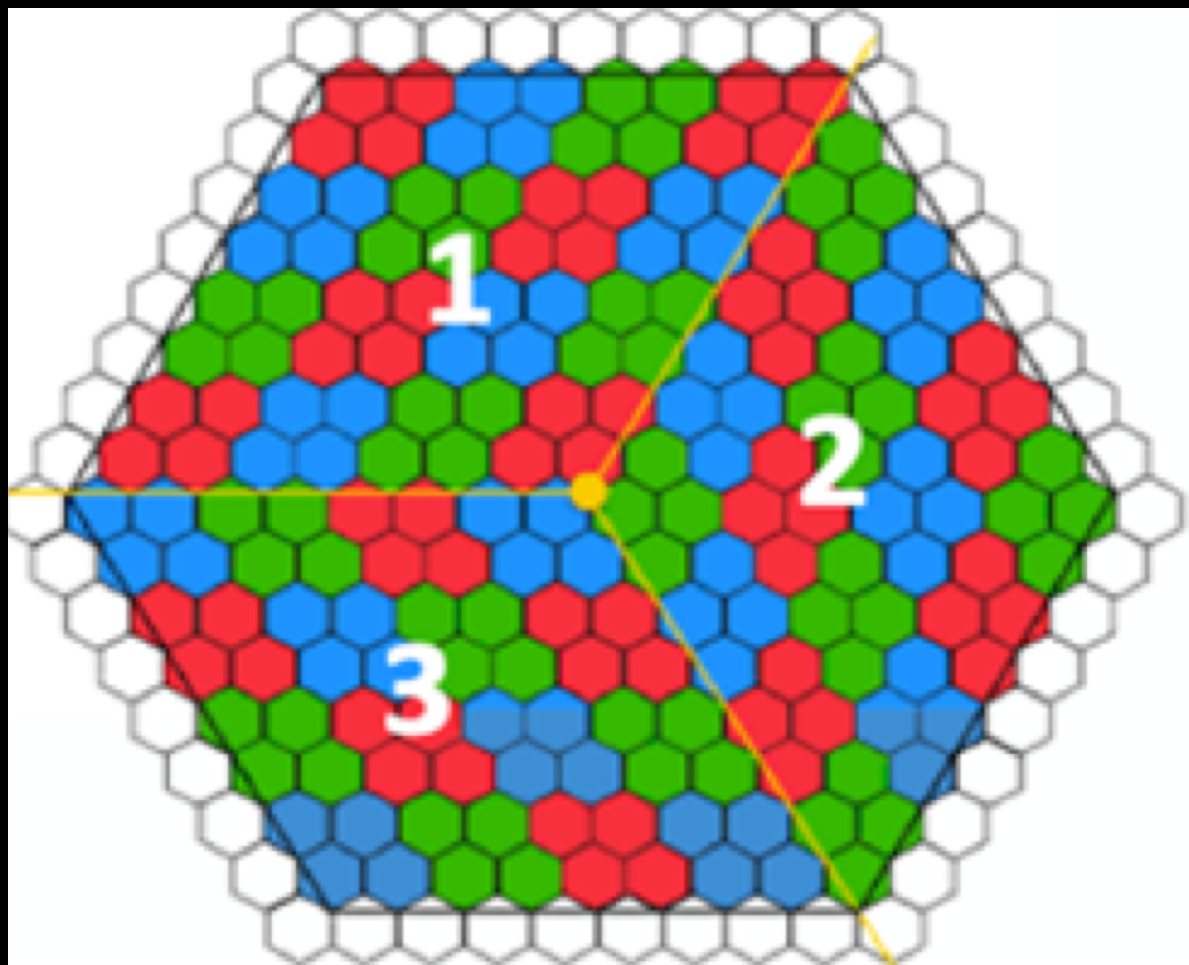
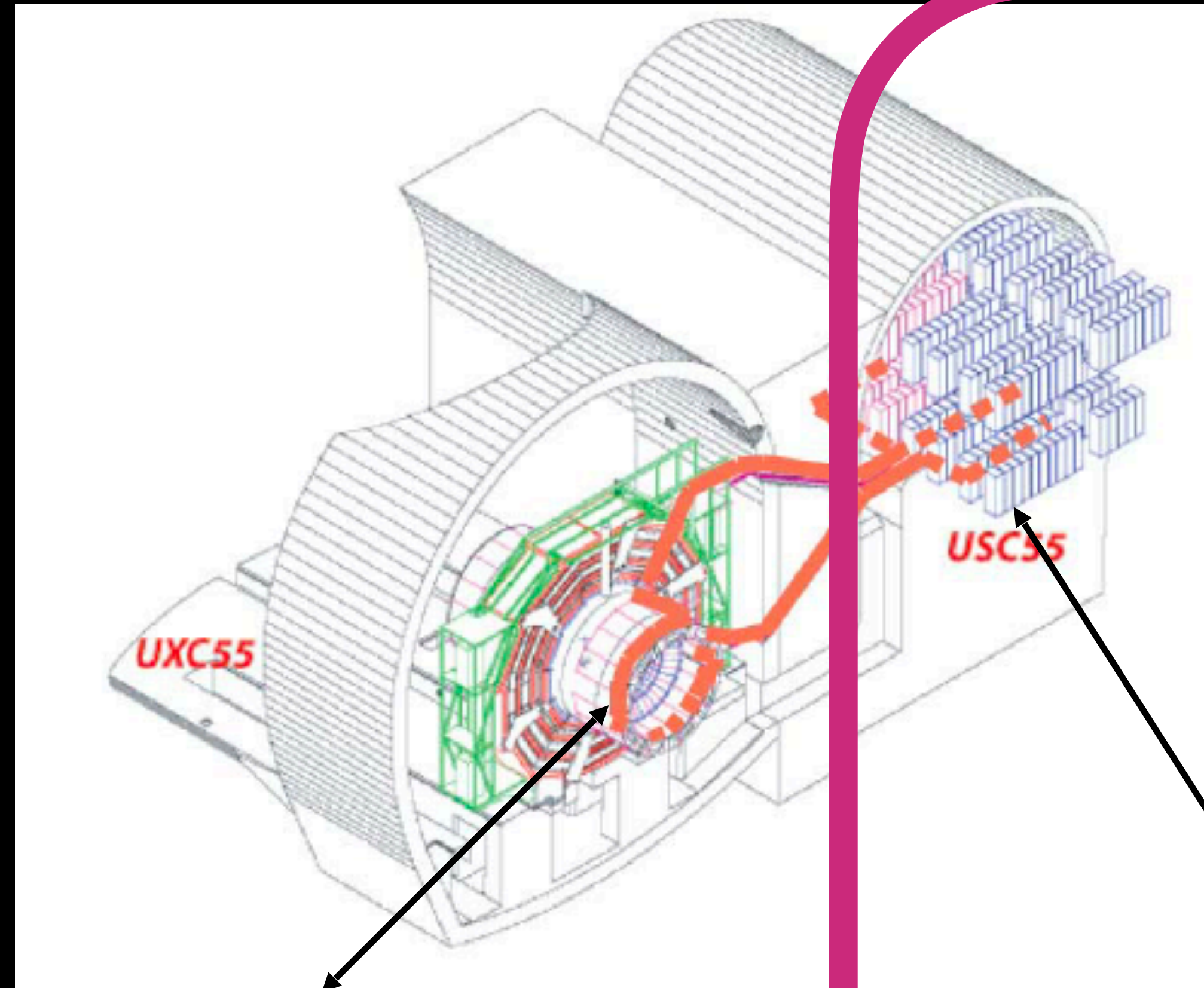
Nanosecond ML inference on FPGAs!

~40 billion inferences/s during HL-LHC



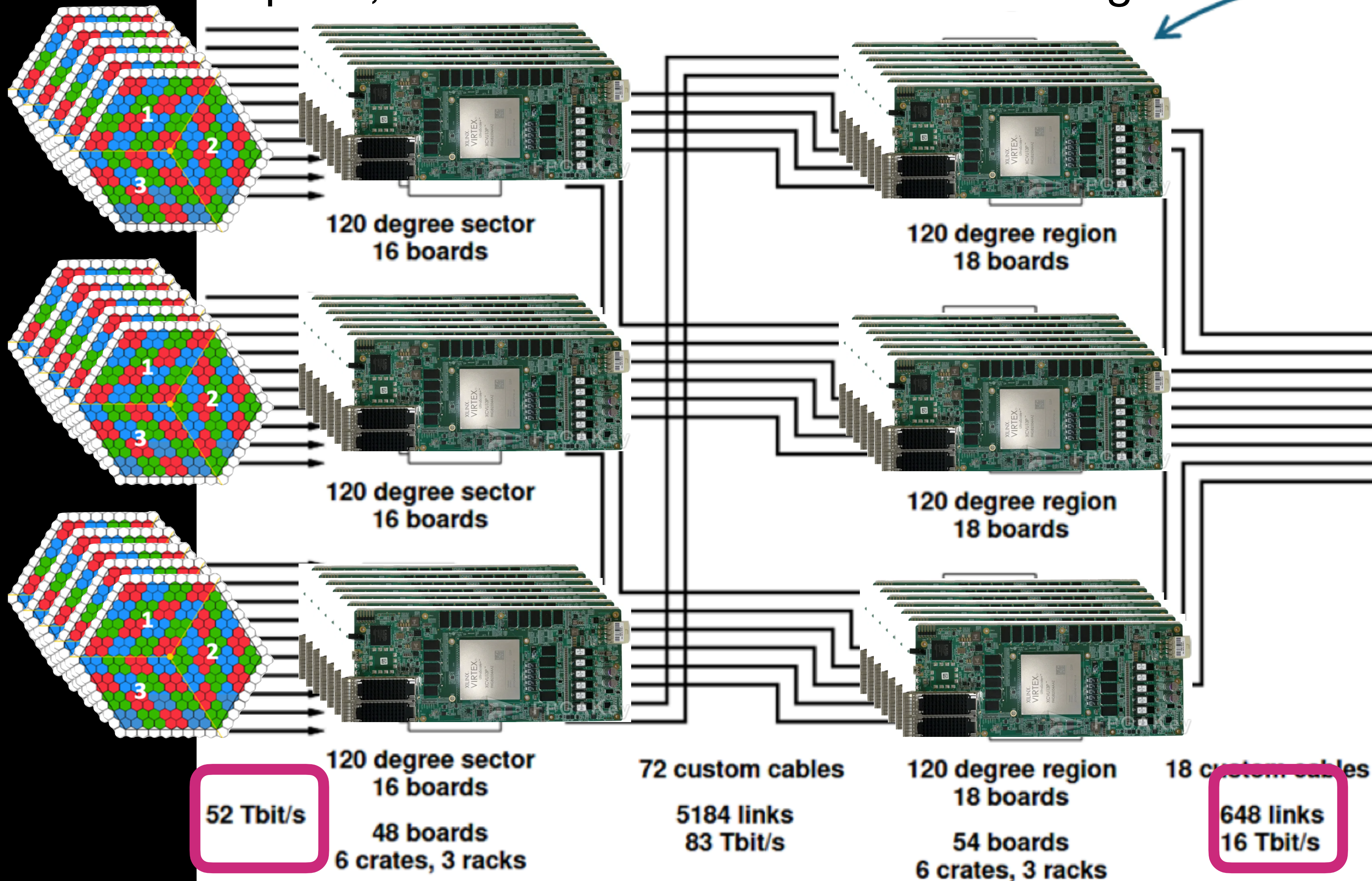
HEP developed
libraries for fast
ML on FPGAs





Stage 1 Unpack, data movement

Stage 2 Clustering



FPGAs process 3 different geometric sectors.

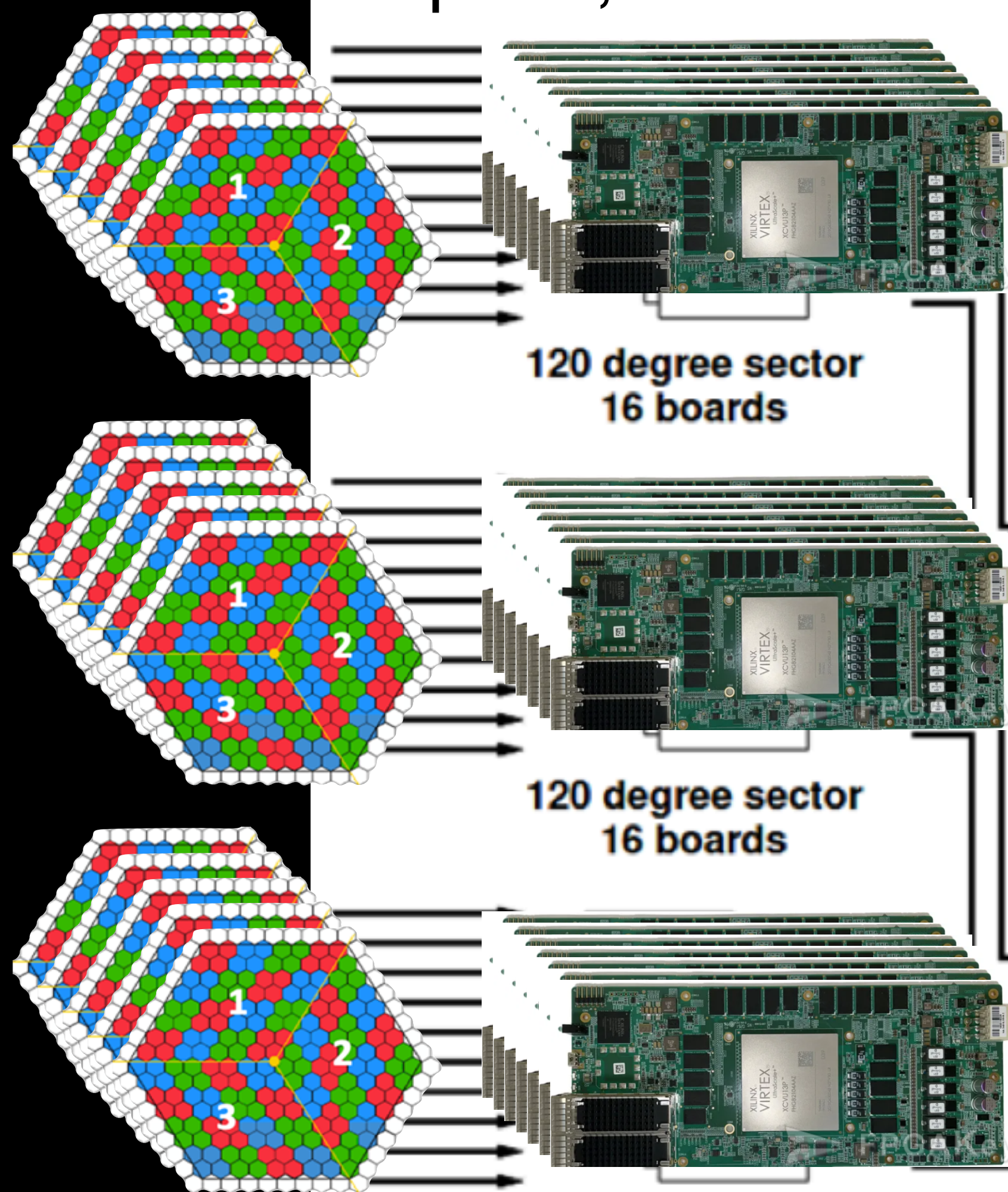
~5,000 “sensor inputs” per FPGA

(and 18 boards to process 18 events in parallel)

Stage2: 54 boards

Lower-level reconstruction tasks

Stage 1 Unpack, data movement



120 degree sector
16 boards

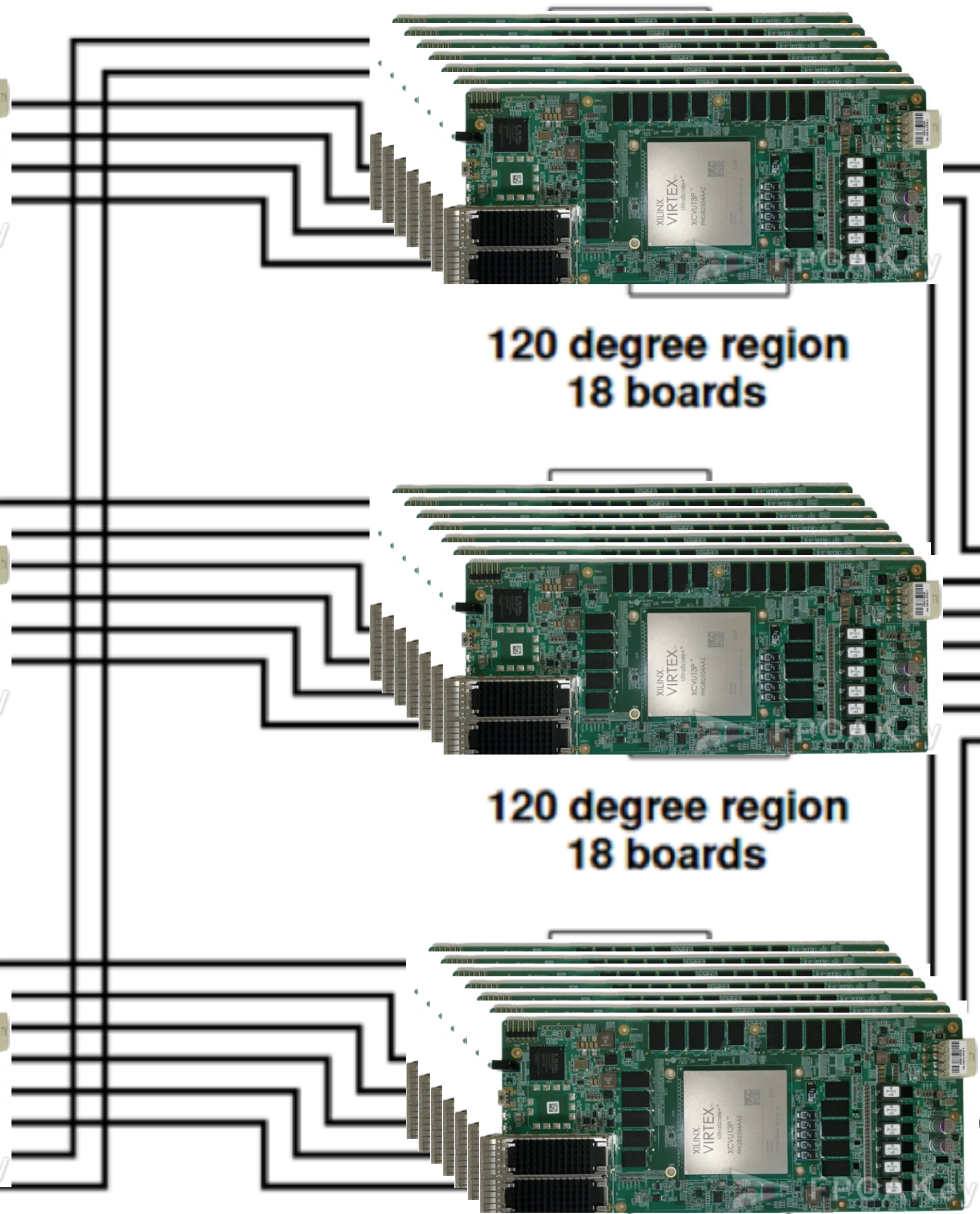
120 degree sector
16 boards

120 degree sector
16 boards

52 Tbit/s

48 boards
6 crates, 3 racks

Stage 2 Clustering



120 degree region
18 boards

120 degree region
18 boards

120 degree region
18 boards

72 custom cables

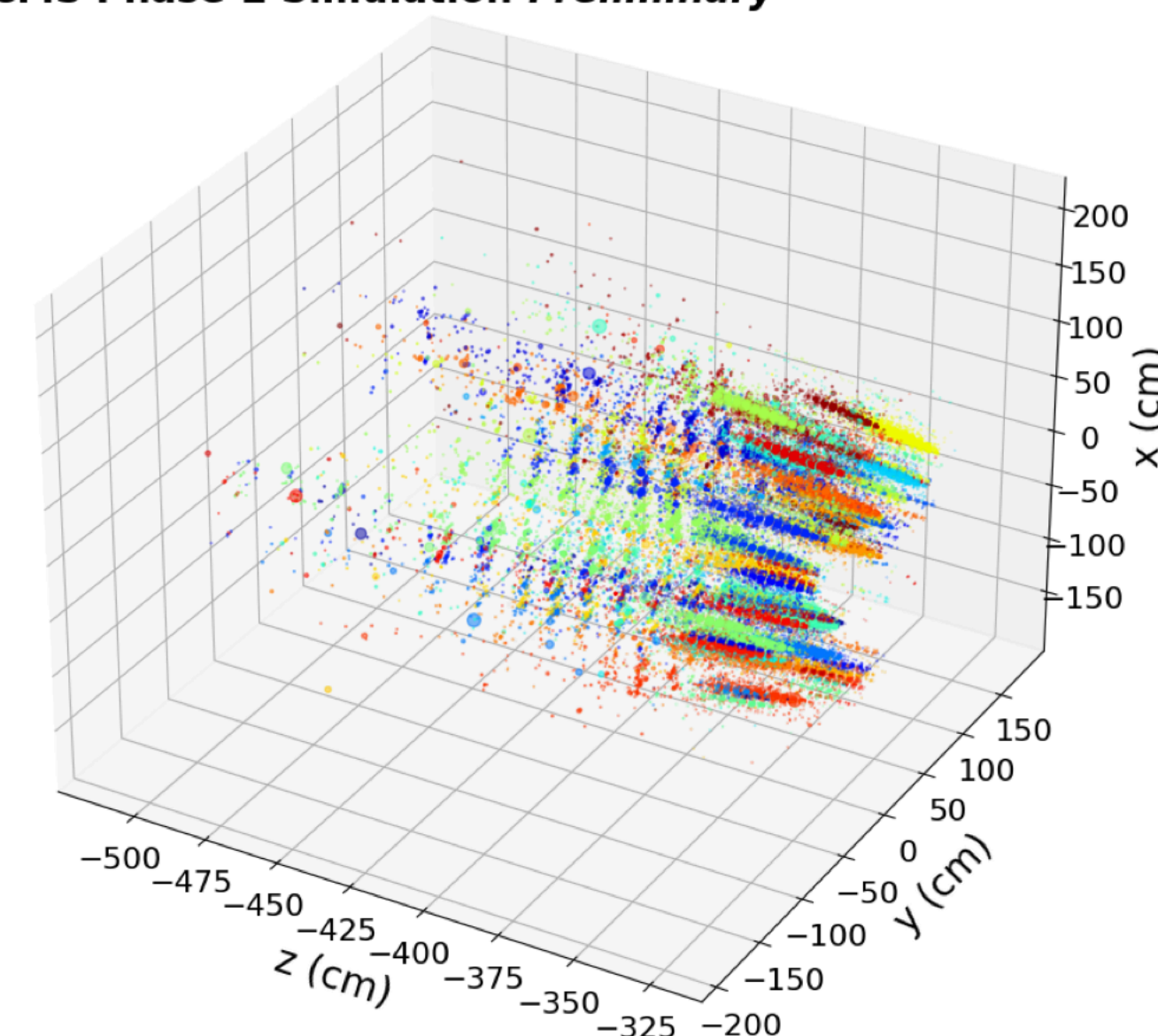
5184 links
83 Tbit/s

18 custom cables

648 links
16 Tbit/s

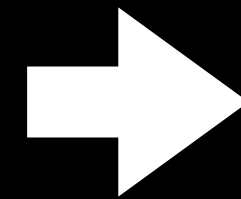
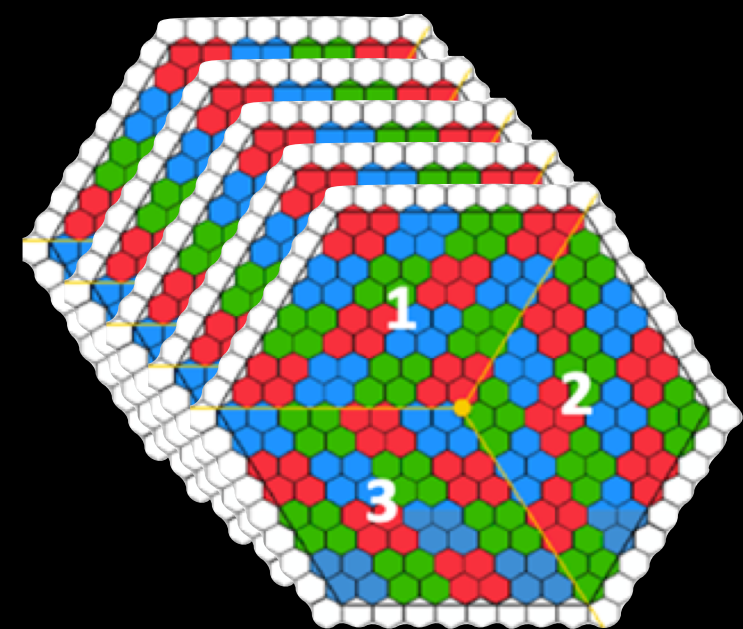
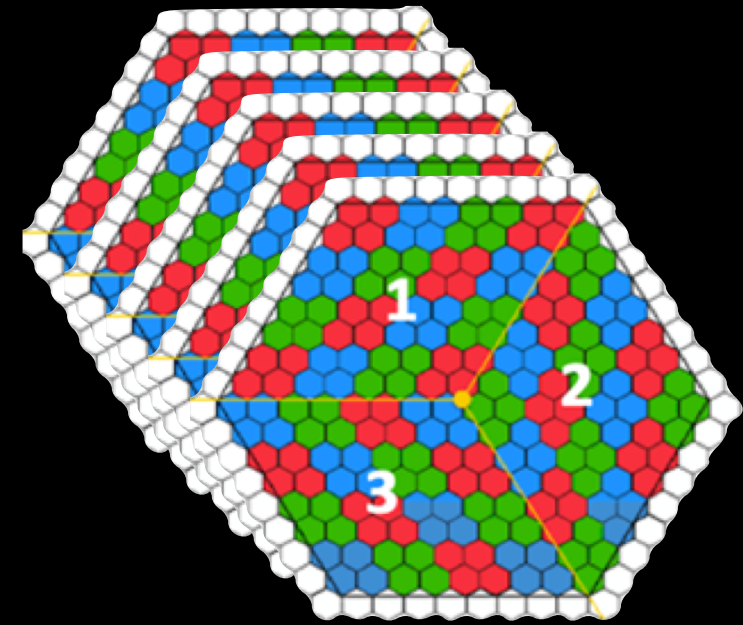
Total 5 μ s to get data out,
cluster into distinct
particle energy clusters
and send downstream

CMS Phase-2 Simulation *Preliminary*



Lower-level reconstruction tasks

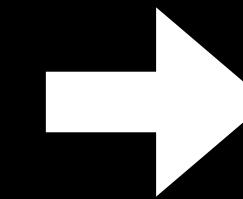
ML models as clustering surrogates: Hits belonging to same object clustered



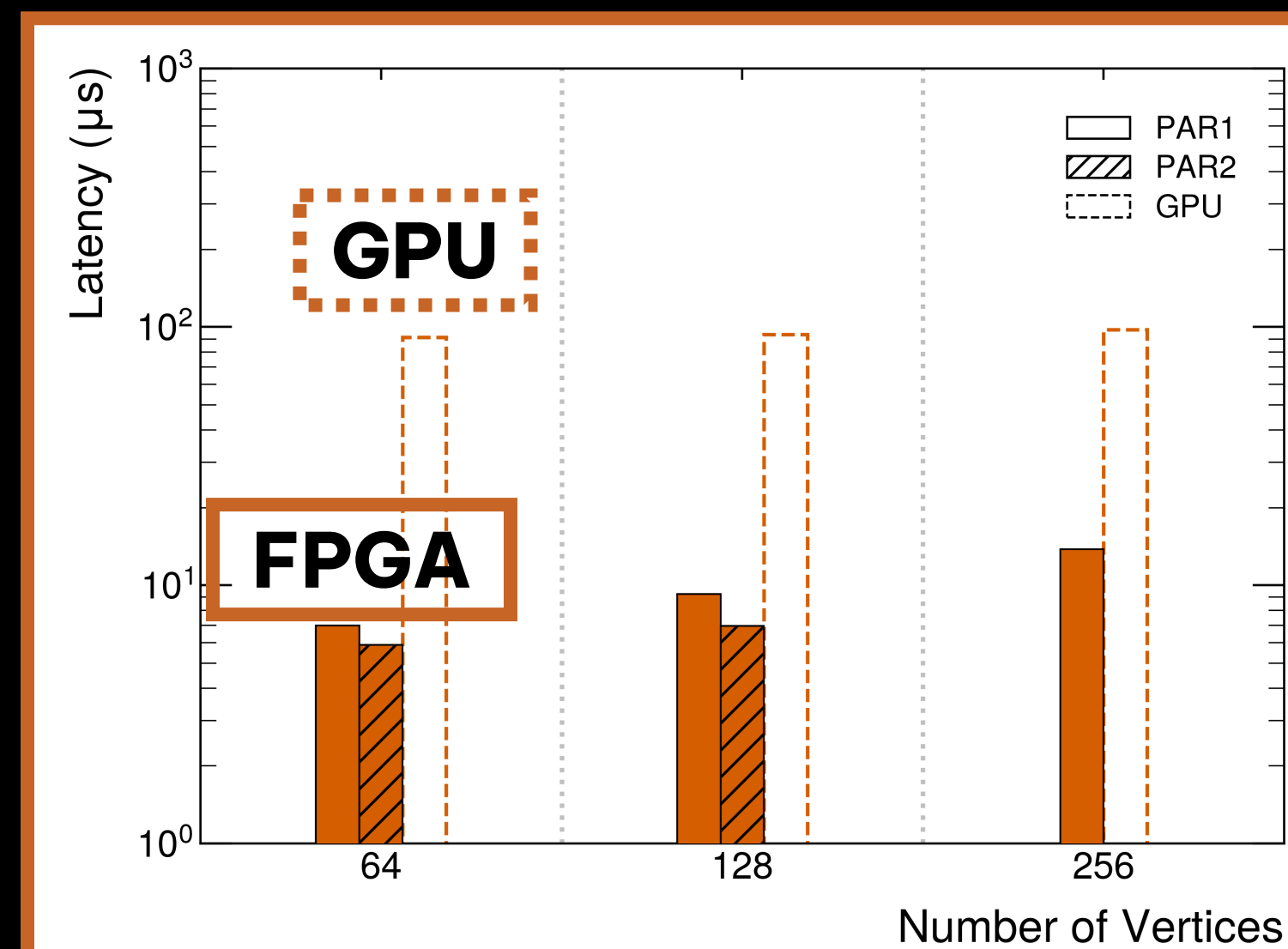
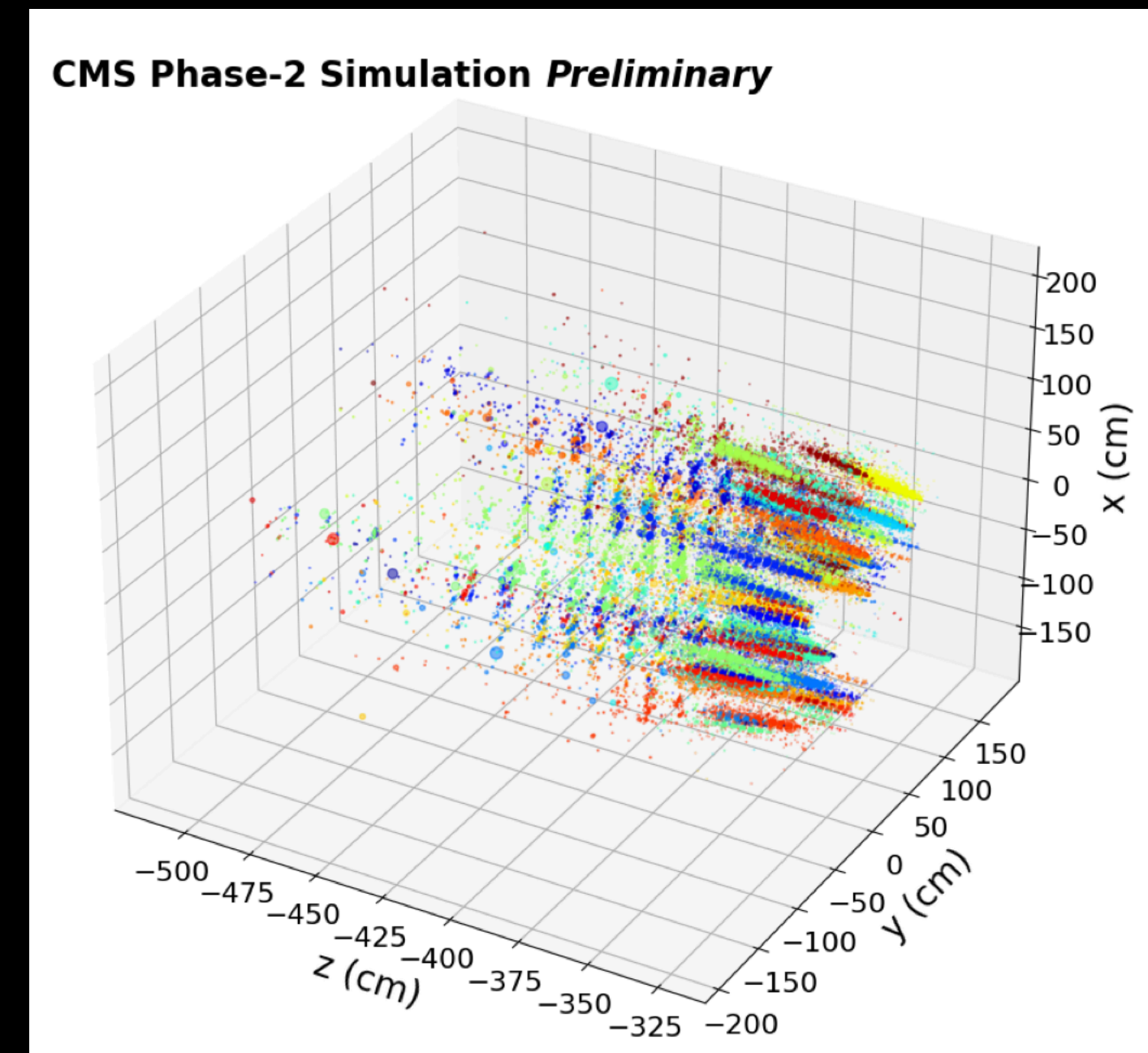
Classical Geometric
Approaches?

Masked Transformers?

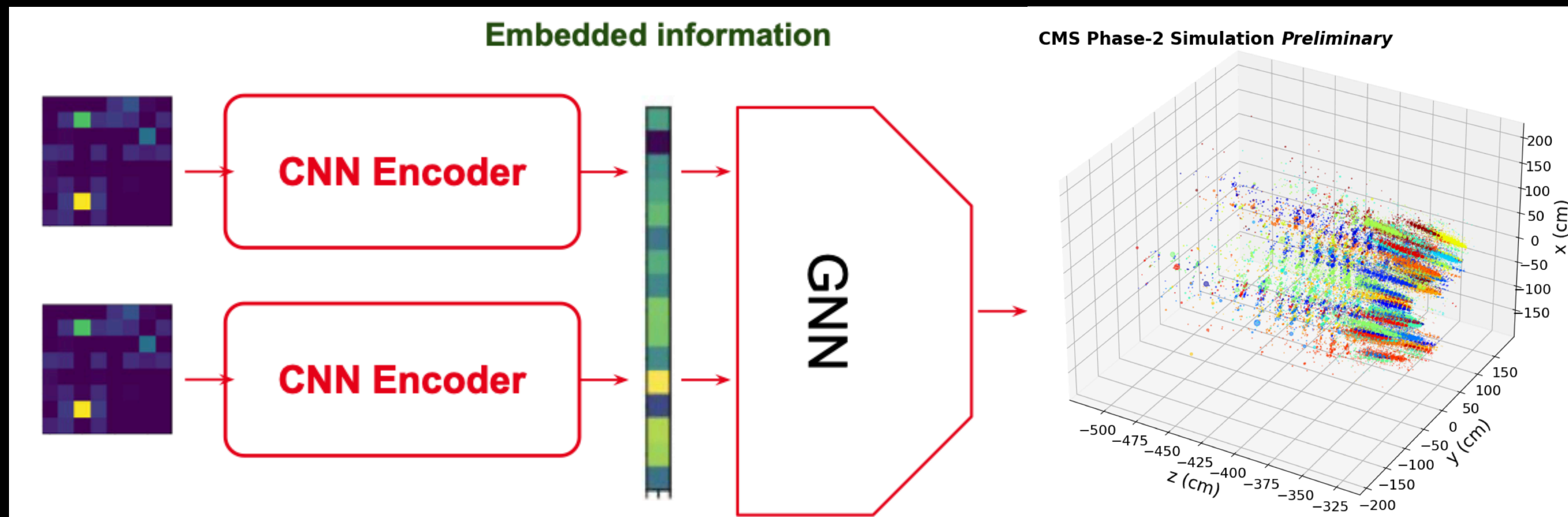
Graph Neural
Networks?



~2–3 μs latency in Stage-2!



E.g GravNet clustering surrogate
L. Asfour



Or one distributed neural network which spans all
~O(100) FPGA boards.

ECON-T NN co-trained with decoder and energy/position regressor!

regularization term for the regression task, co-optimised with front-end NN!

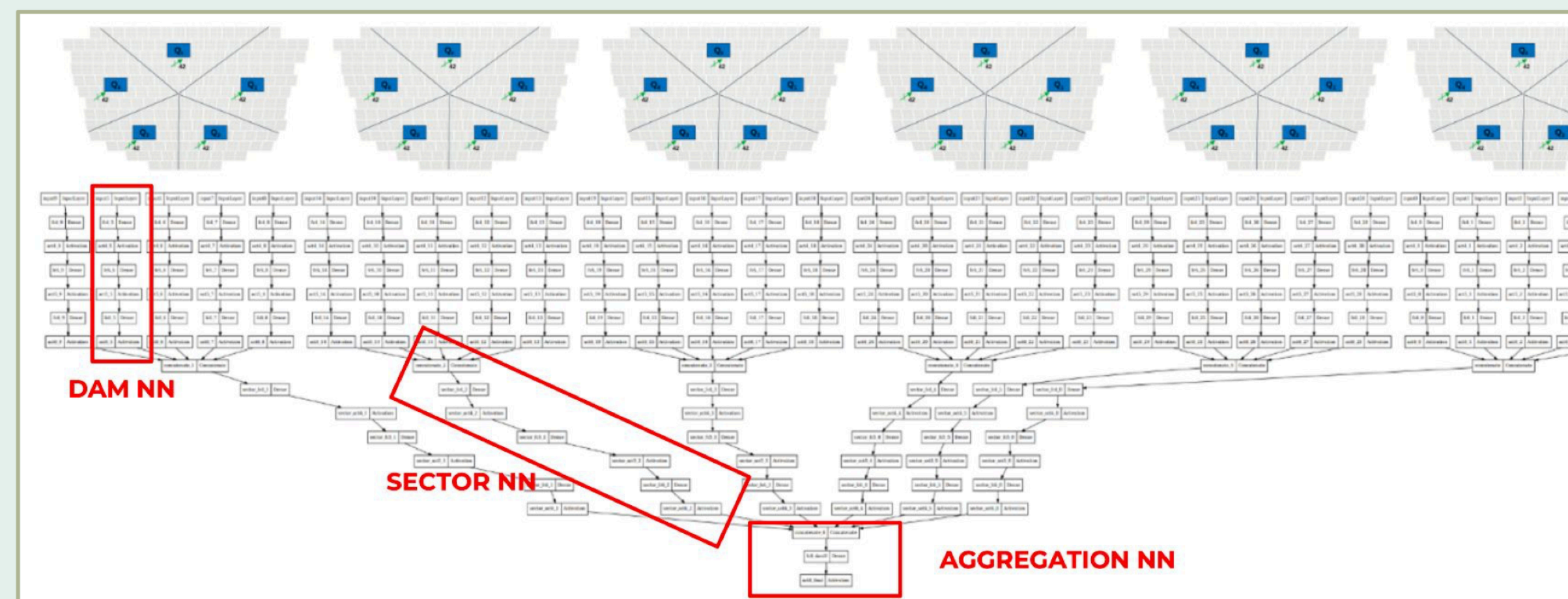
DRIFTS

Paul Devougue, Ozgur
Sahin

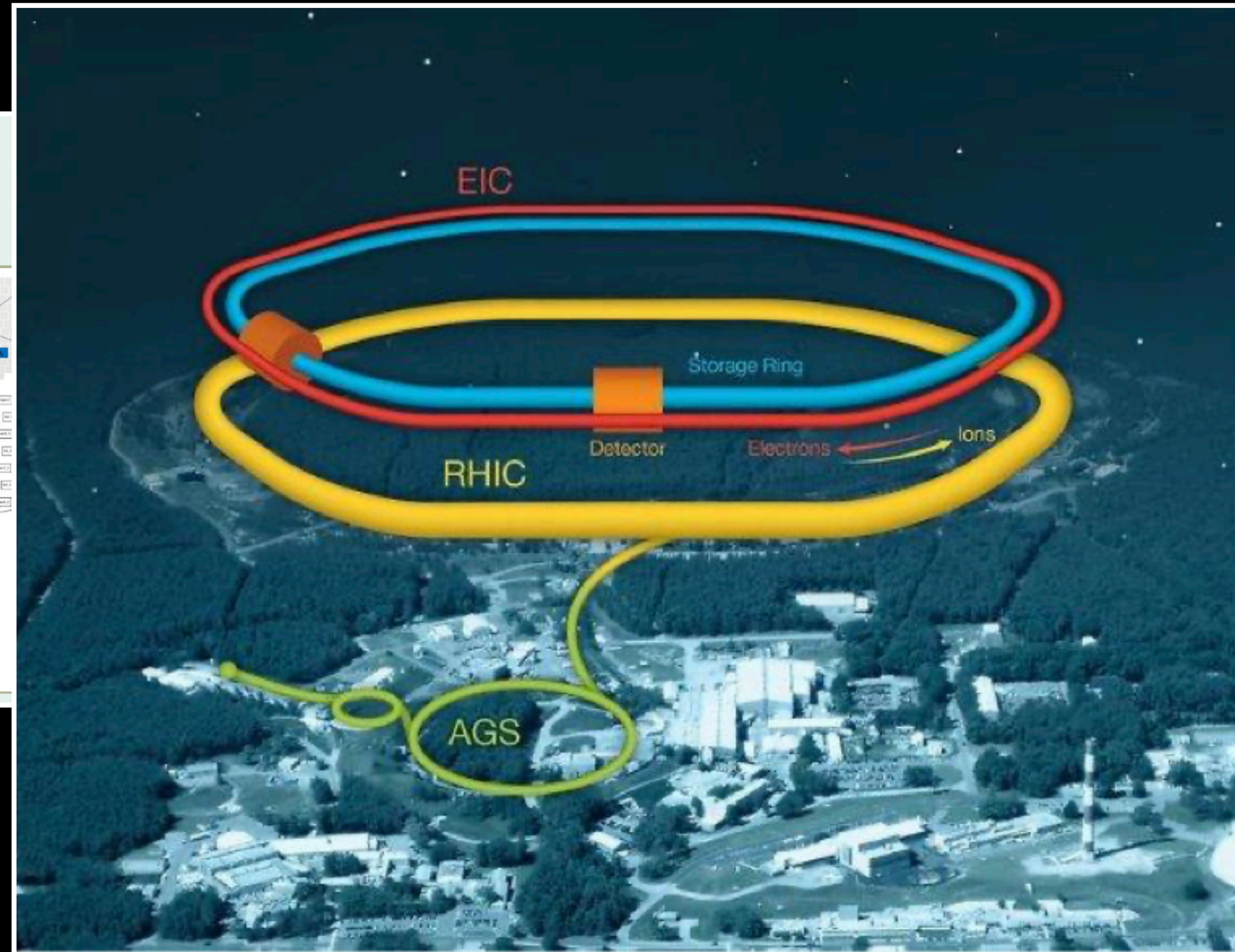
Multi-FPGA NNs for real time data reduction at ePIC at the EIC

The NN model mimics the DAQ system architecture

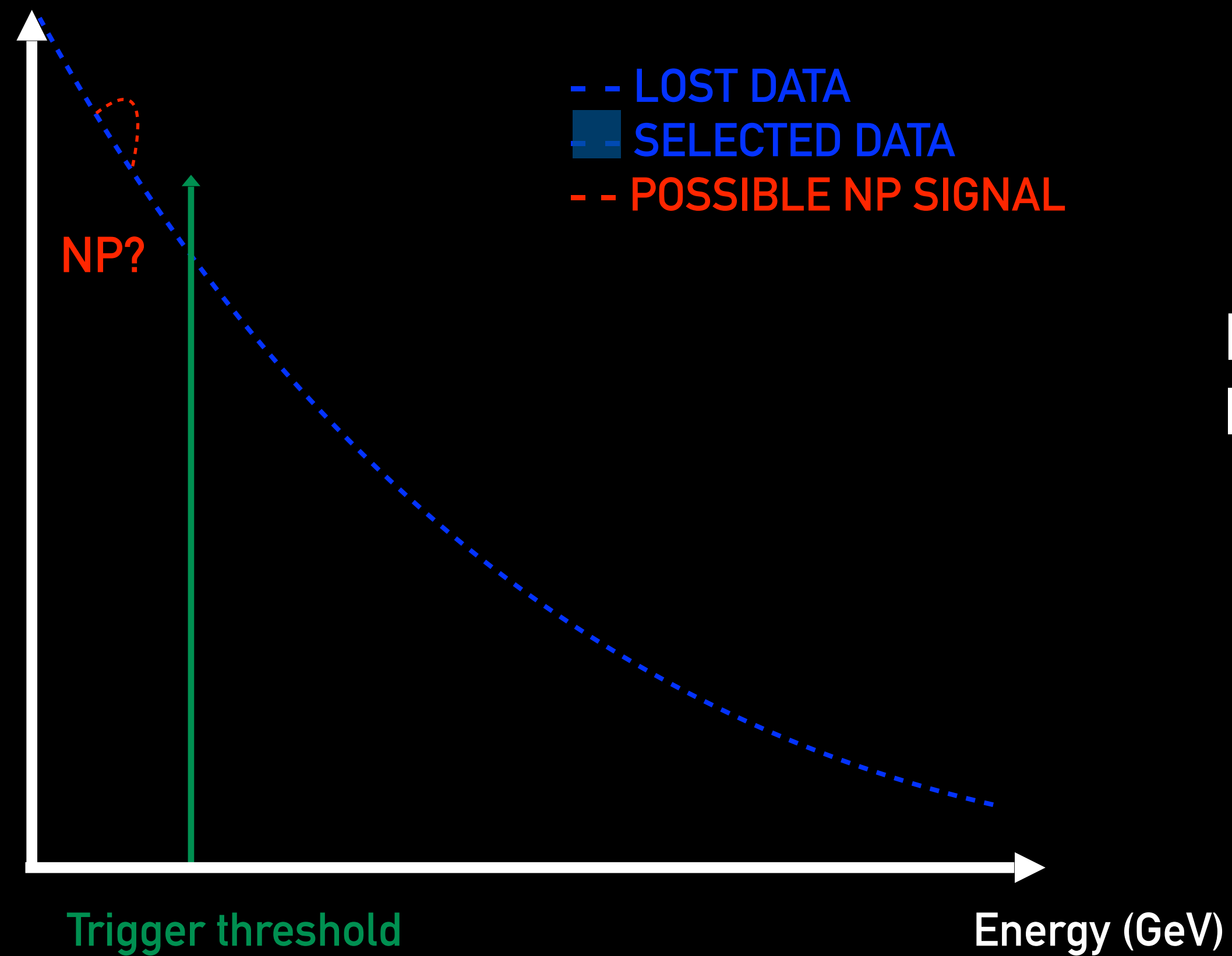
- **30** (# of subsectors x # of sectors) **DAM MLP networks** deployed on 30 **DAM FPGAs**.
- **6 sector MLP networks + 1 aggregation network** deployed on the **TP FPGA**.



HGQ2+hls4ml
Rossi et. al.



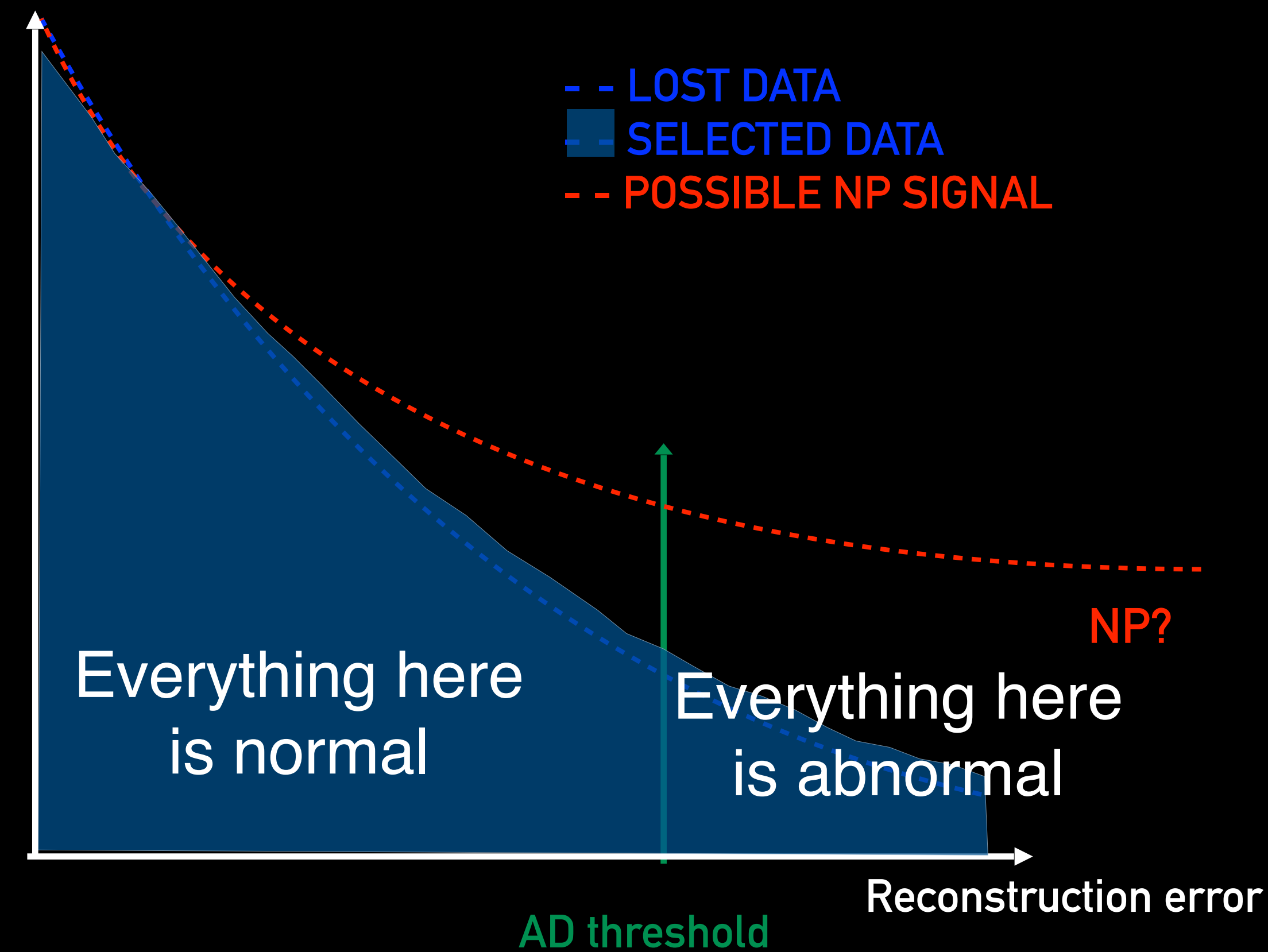
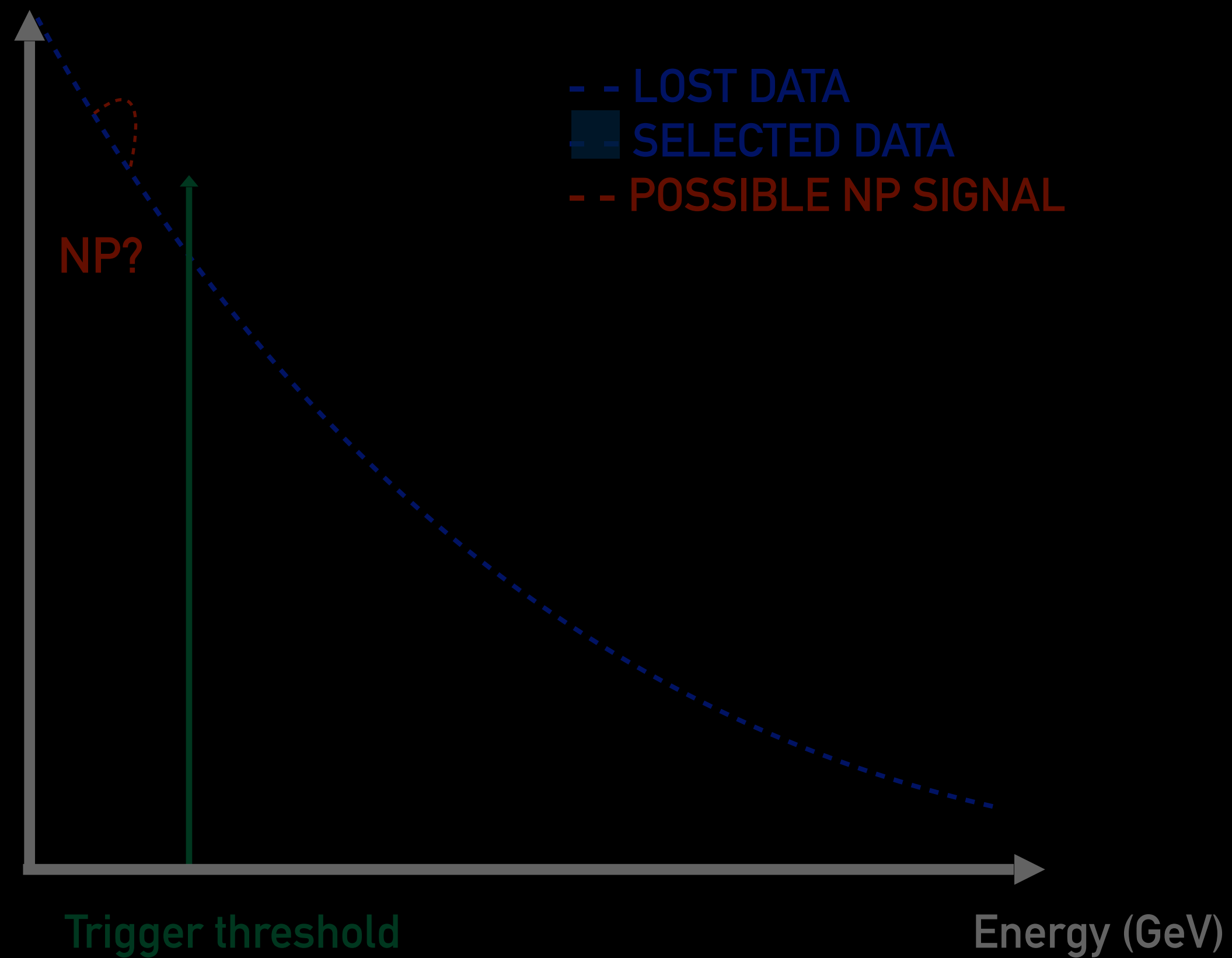
Anomaly Detection triggers



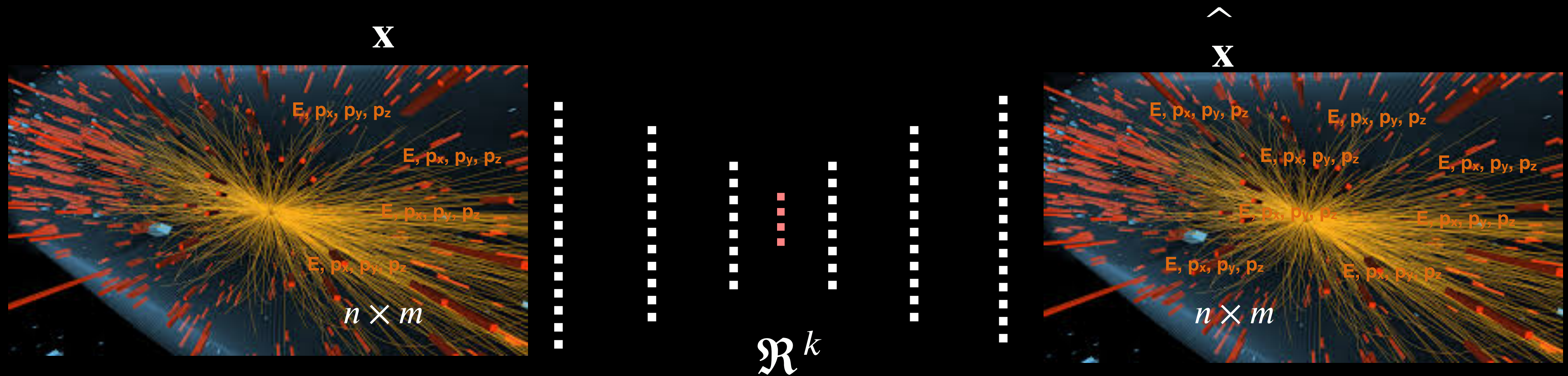
Level-1 rejects >99% of events!
Is there a smarter way to select?

Higher-level decision making tasks

Anomaly Detection triggers

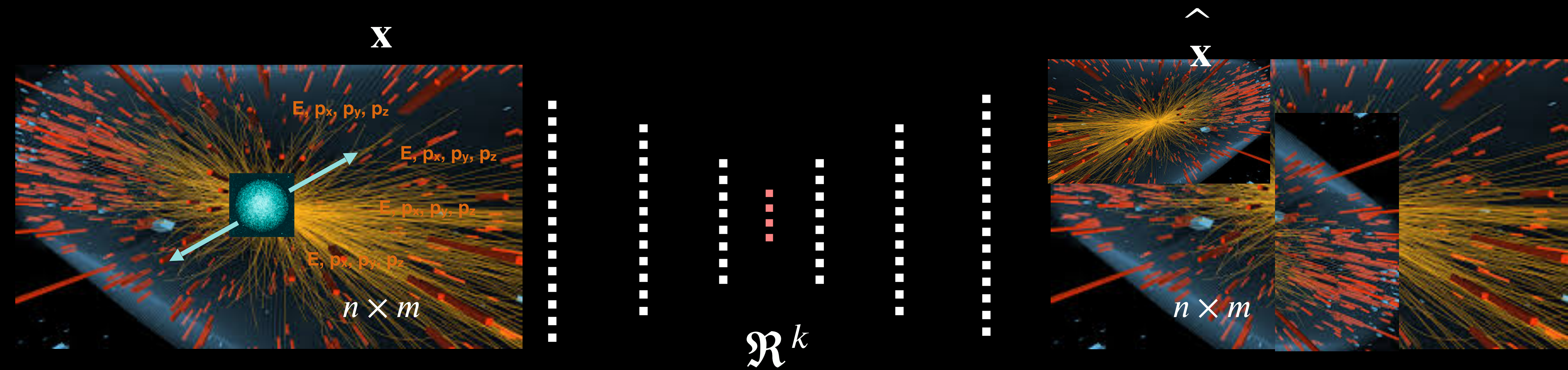


Outlier detection

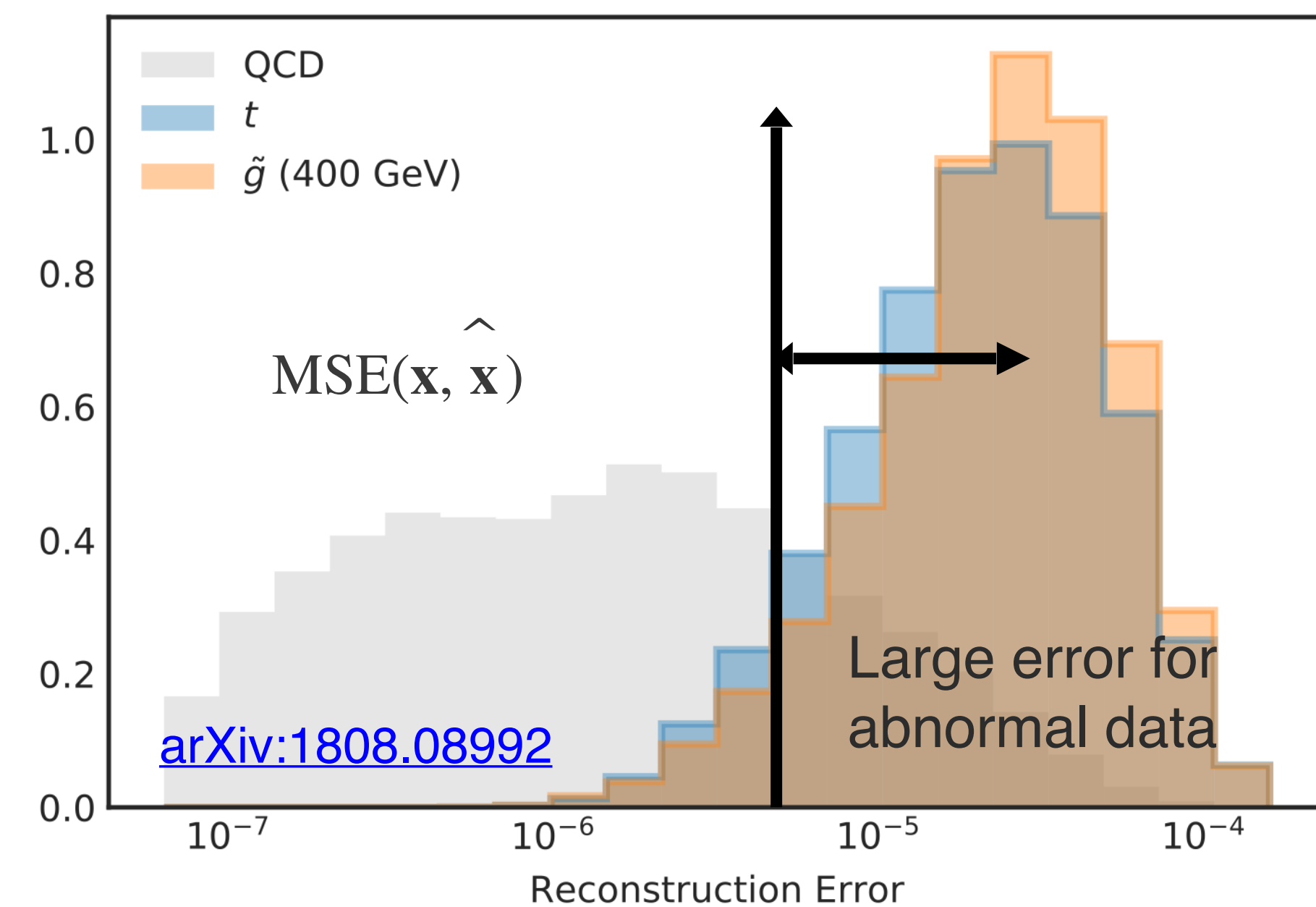
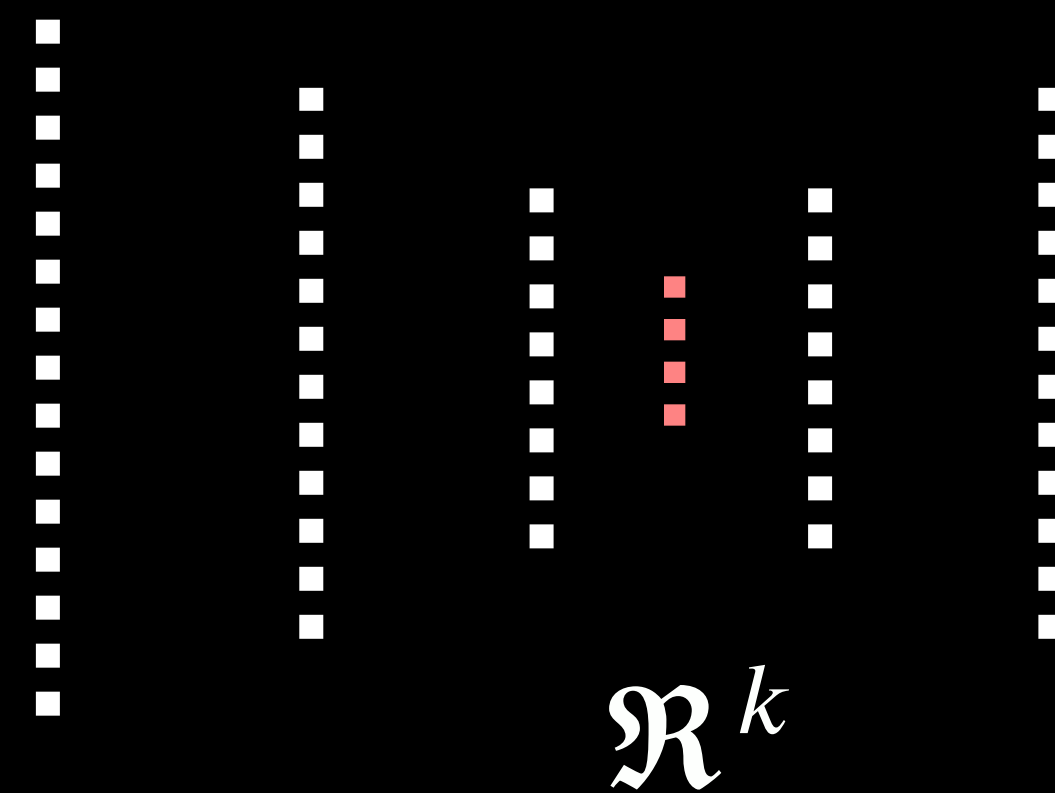
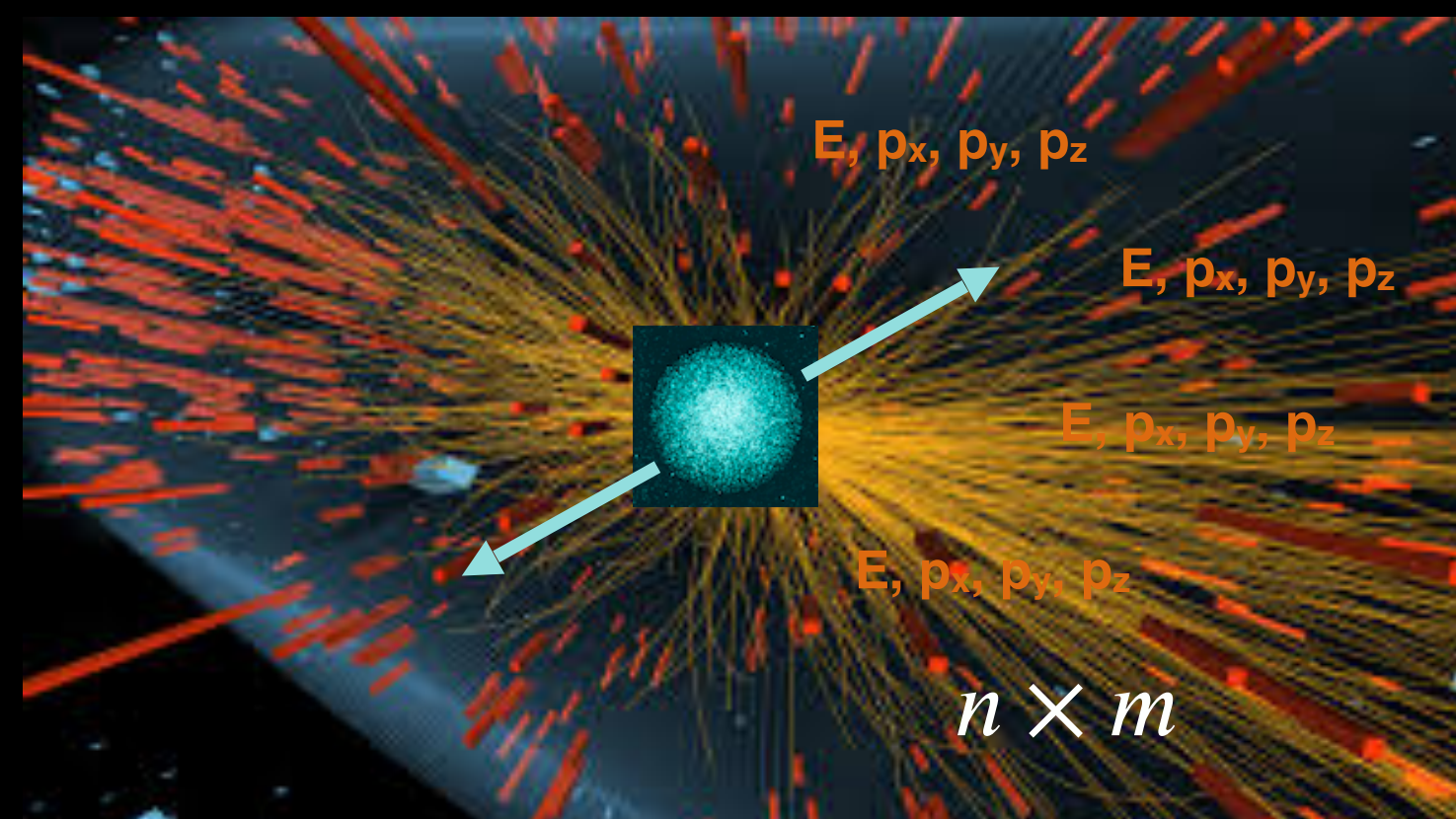


Compressed representation of $\mathbf{x}$.
Latent space $\mathcal{R}^k$, $k < m \times n$
prevents memorisation of input, must learn

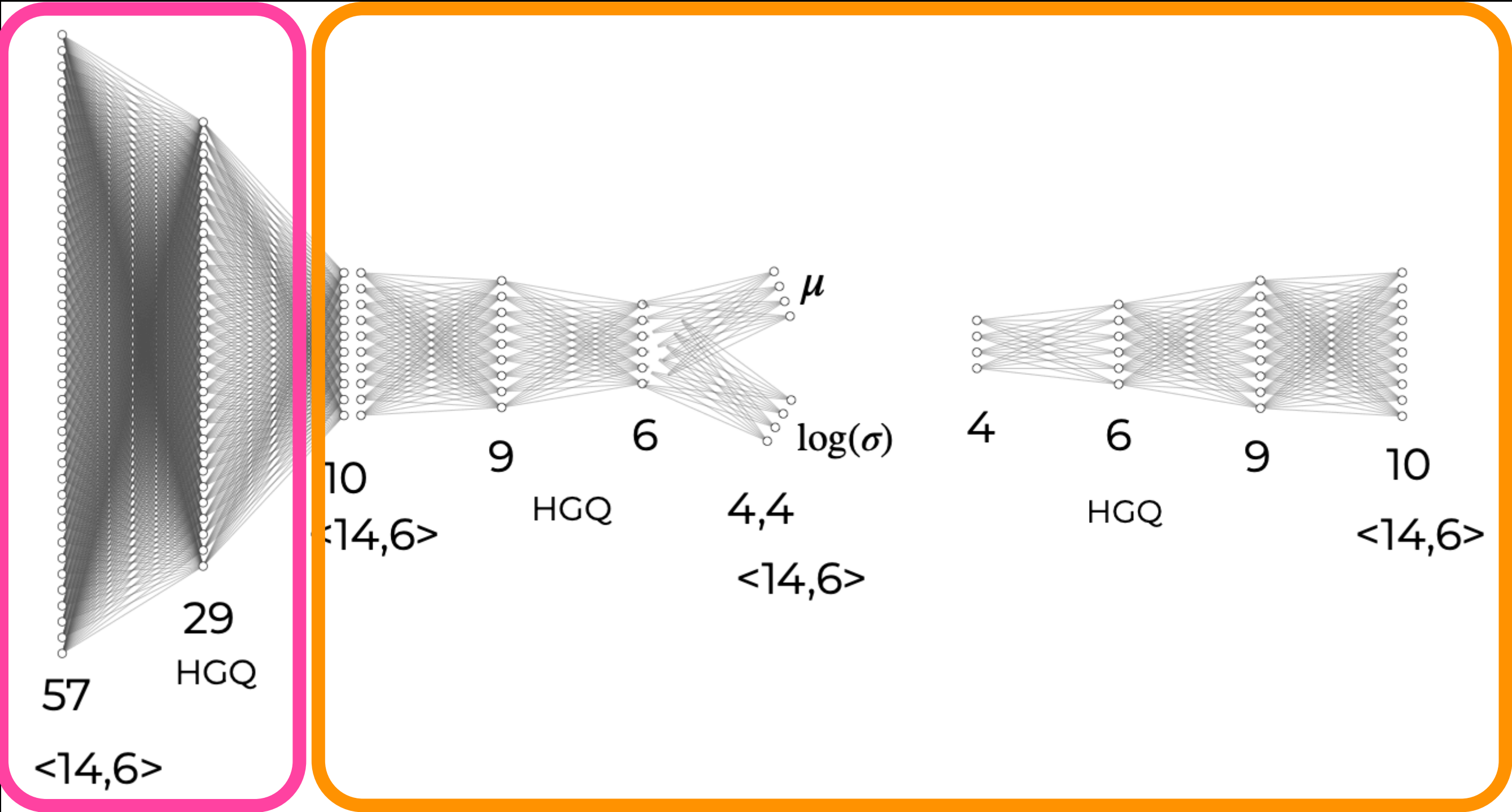
Outlier detection



$\mathcal{L}(\mathbf{X}, \hat{\mathbf{X}})$ is Mean Squared Error($\mathbf{X}, \hat{\mathbf{X}}$), “high error events” proxy for “degree of abnormality”

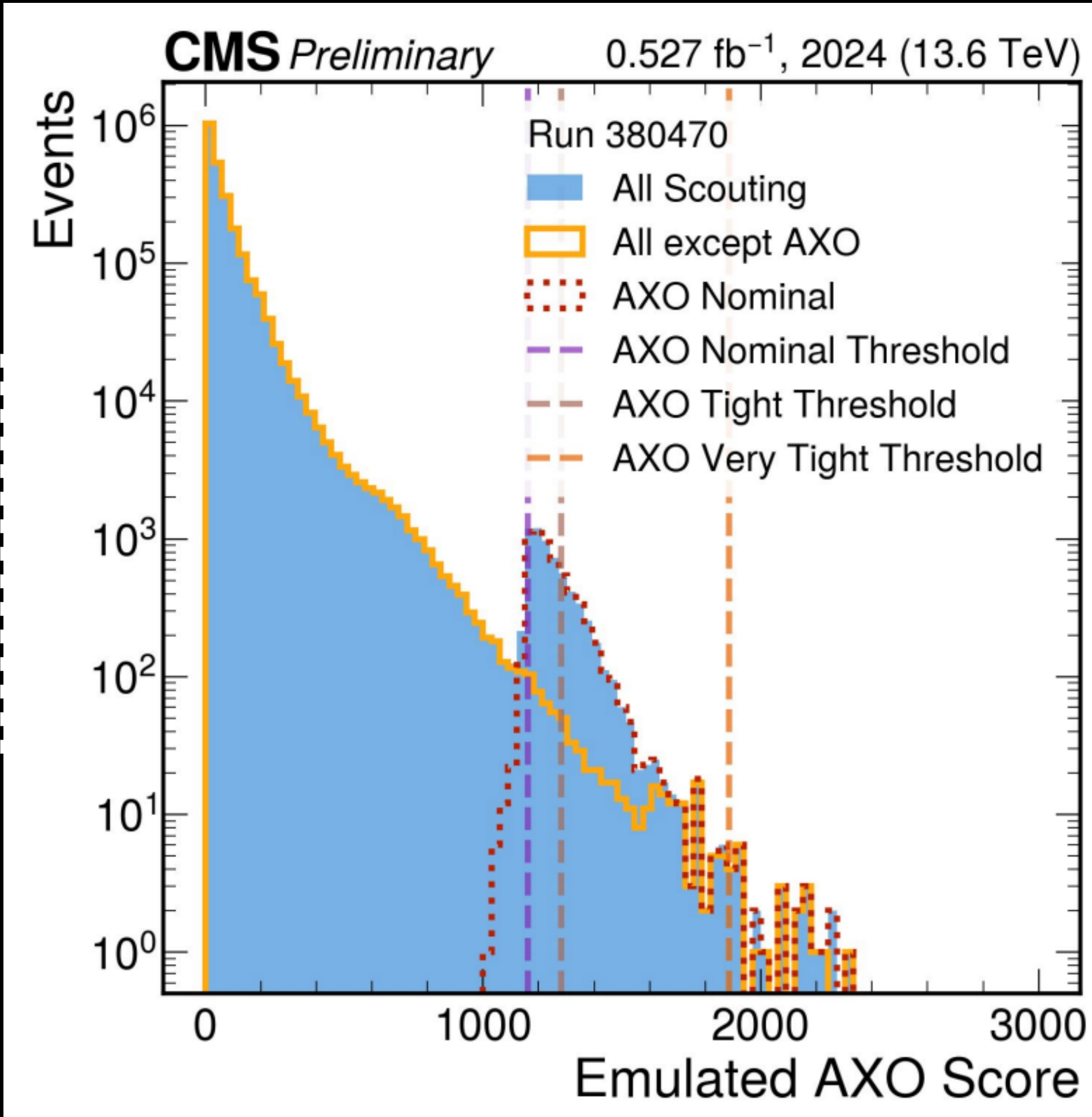


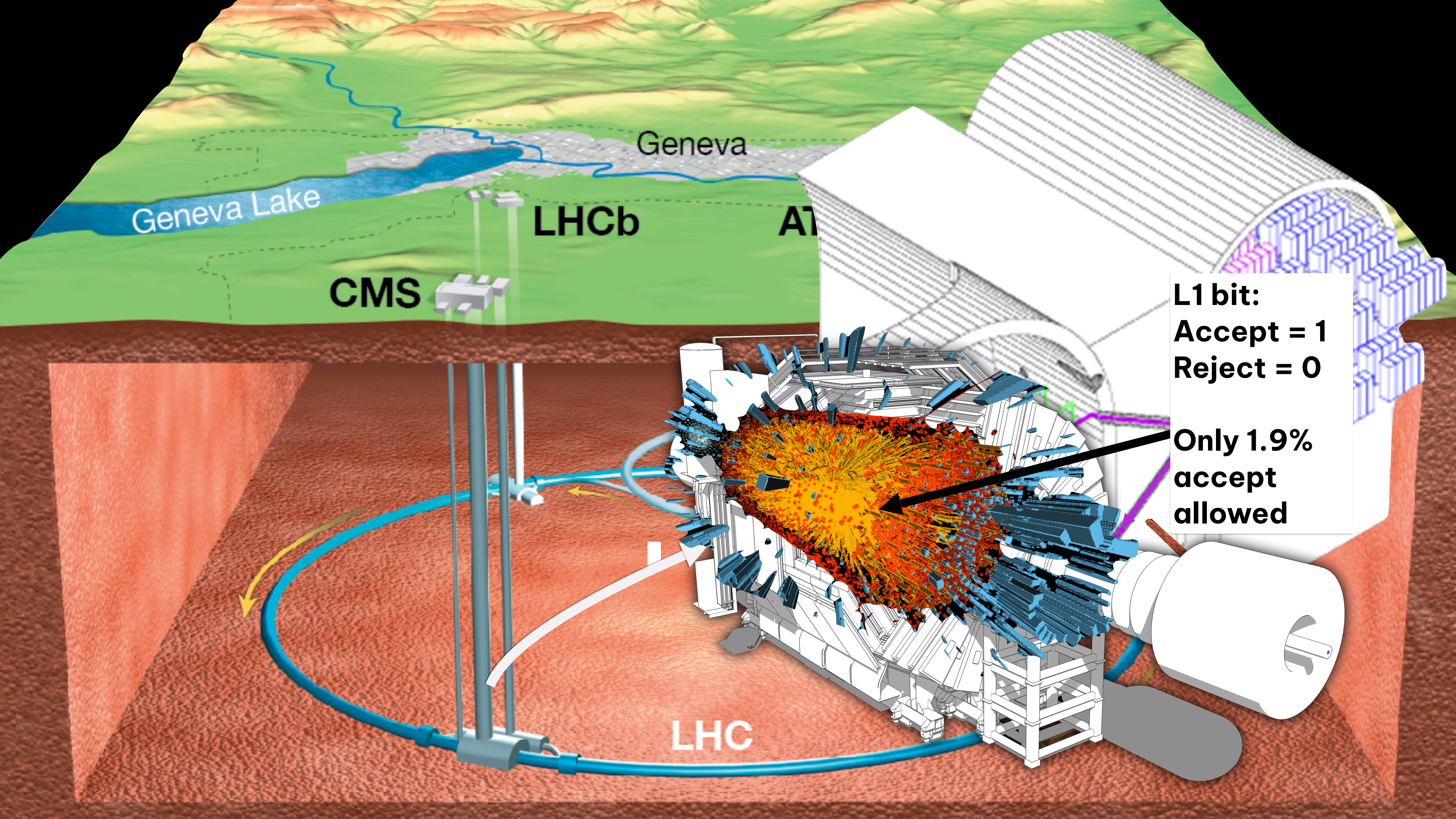
Neural embedding with
VicReg contrastive learning



Variational Autoencoder

8 layers in 50 ns on FPGA,
~1% of total rate selected by AD





Geneva

Geneva Lake

LHCb

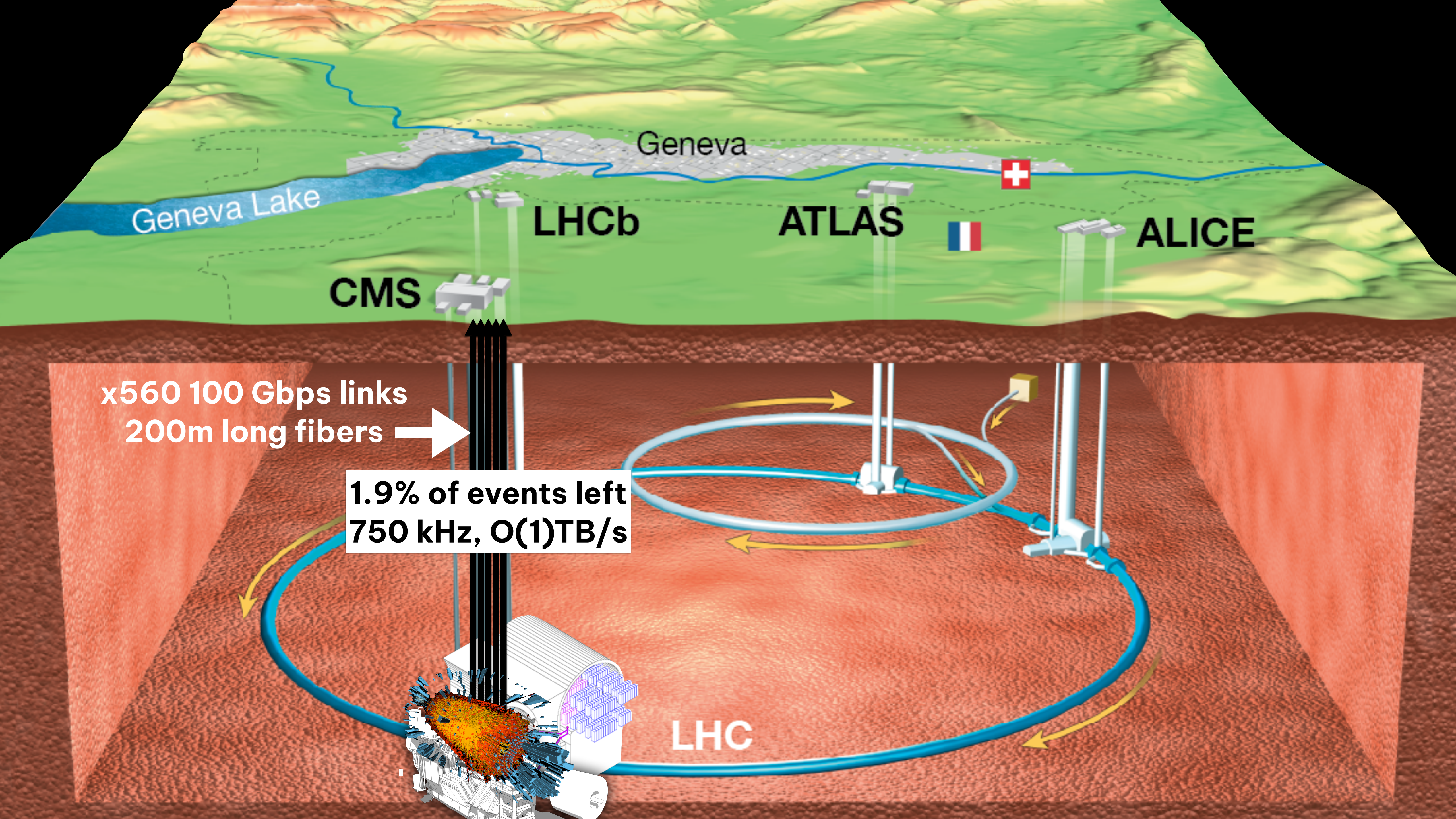
ATLAS

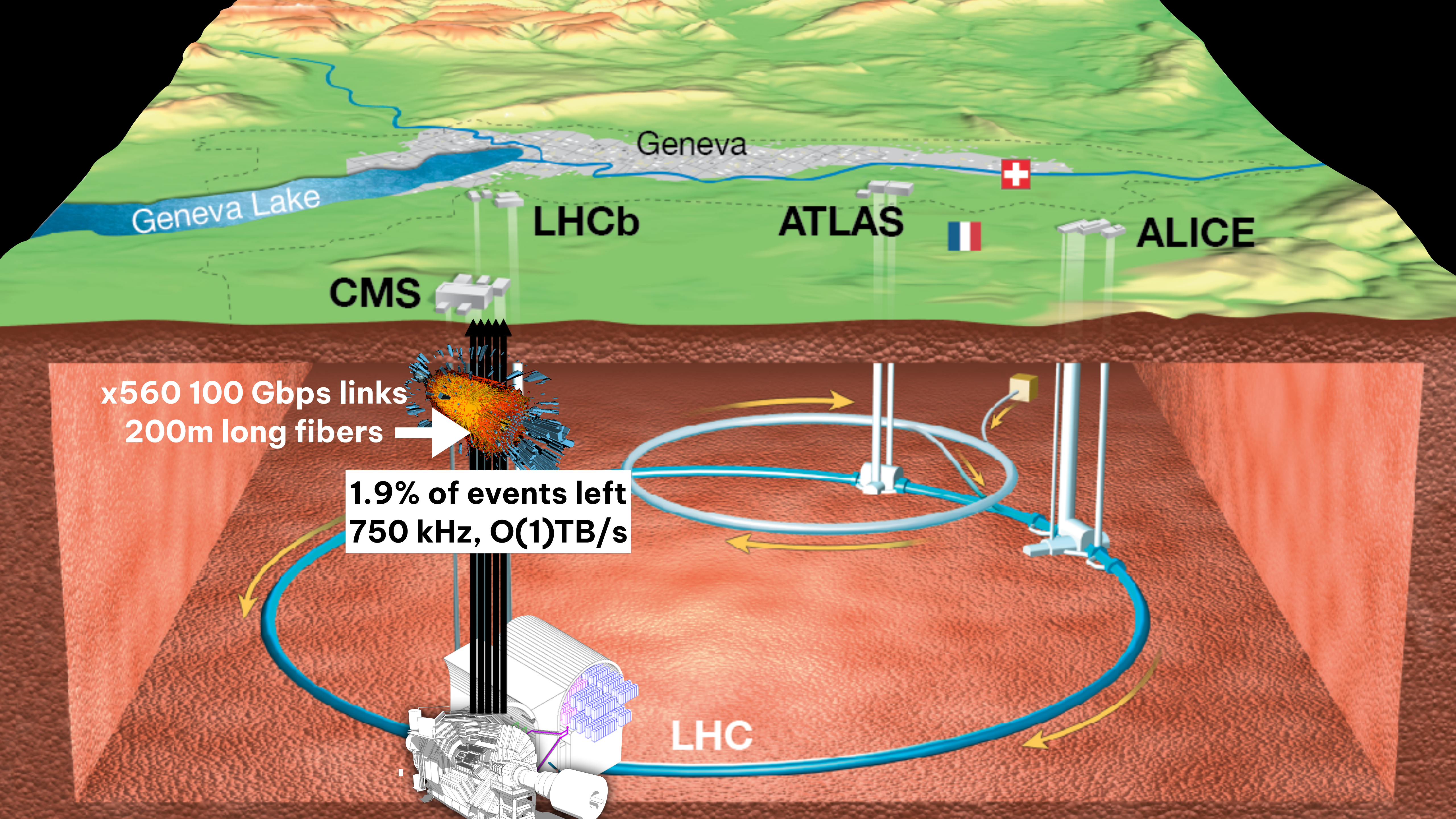
CMS

LHC

L1 bit:
Accept = 1
Reject = 0

Only 1.9%
accept
allowed





Geneva Lake

Geneva

CMS

LHCb

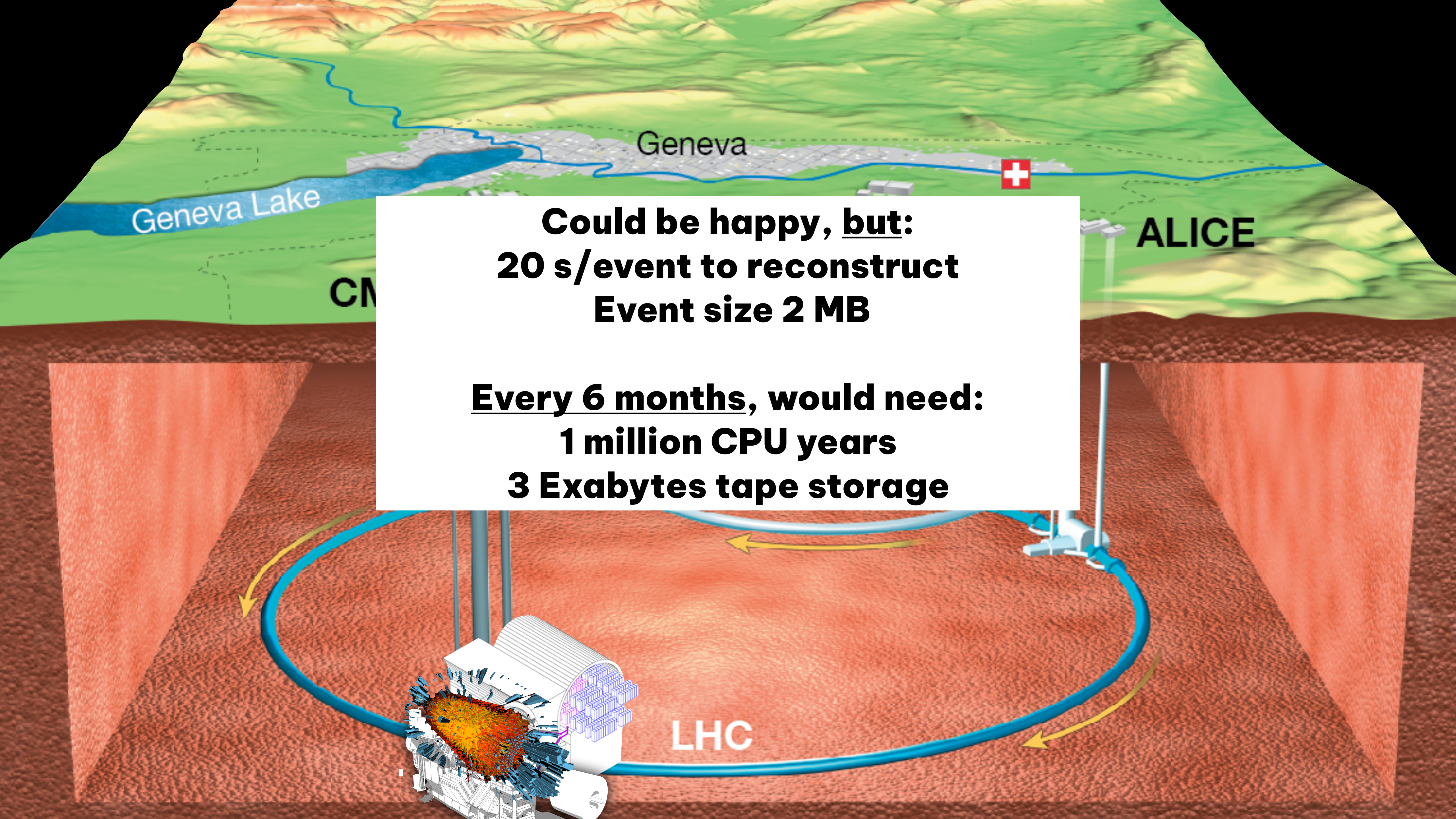
ATLAS

ALICE

x560 100 Gbps links
200m long fibers

1.9% of events left
750 kHz, O(1)TB/s

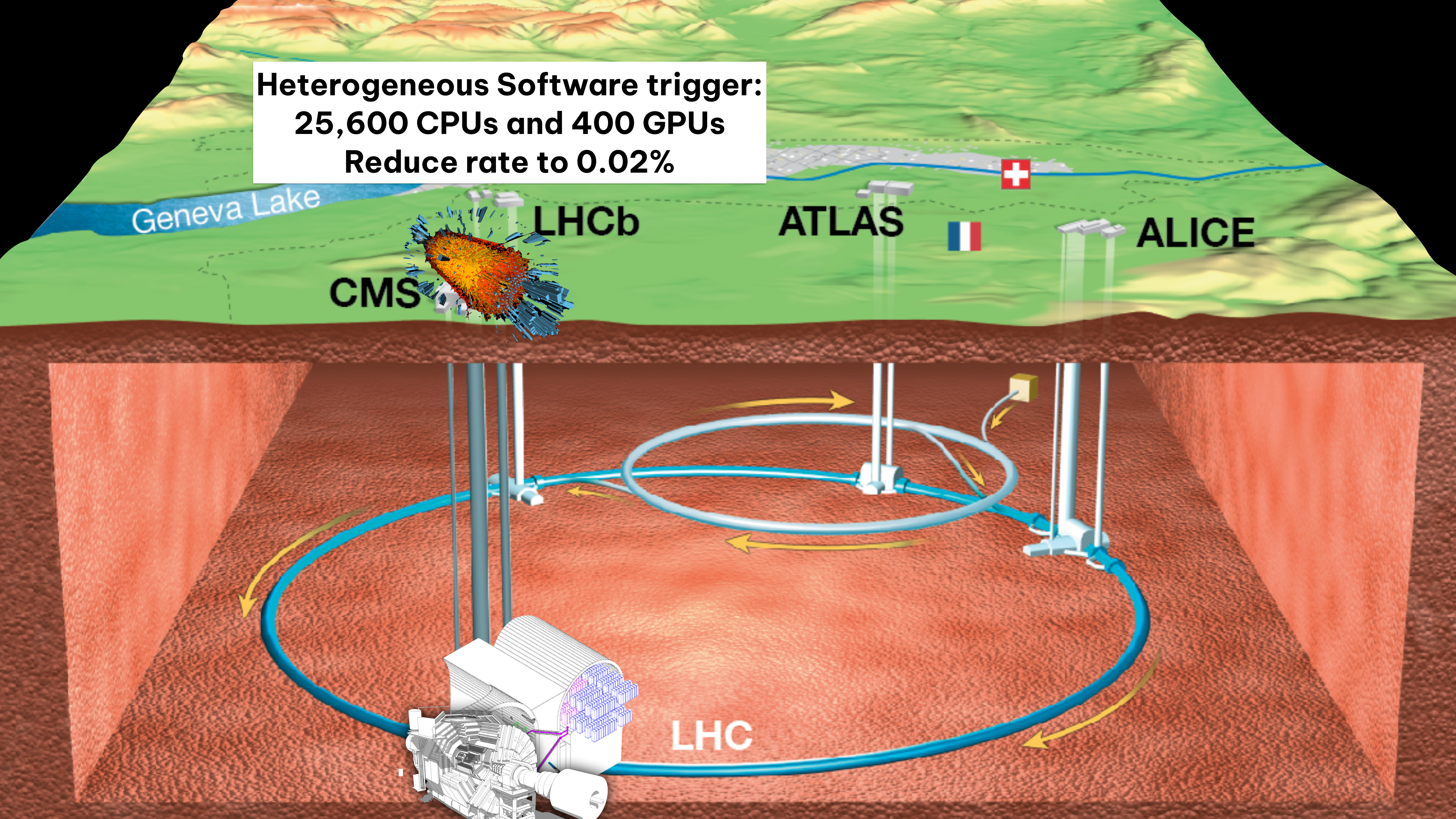
LHC

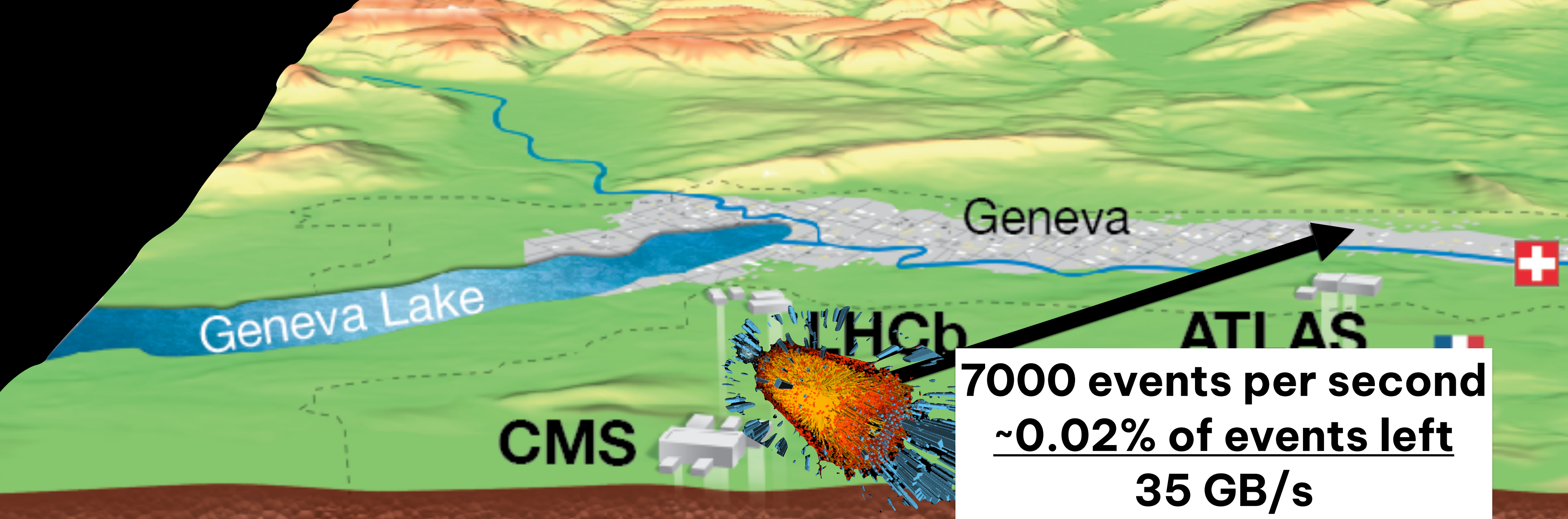


Could be happy, but:
20 s/event to reconstruct
Event size 2 MB

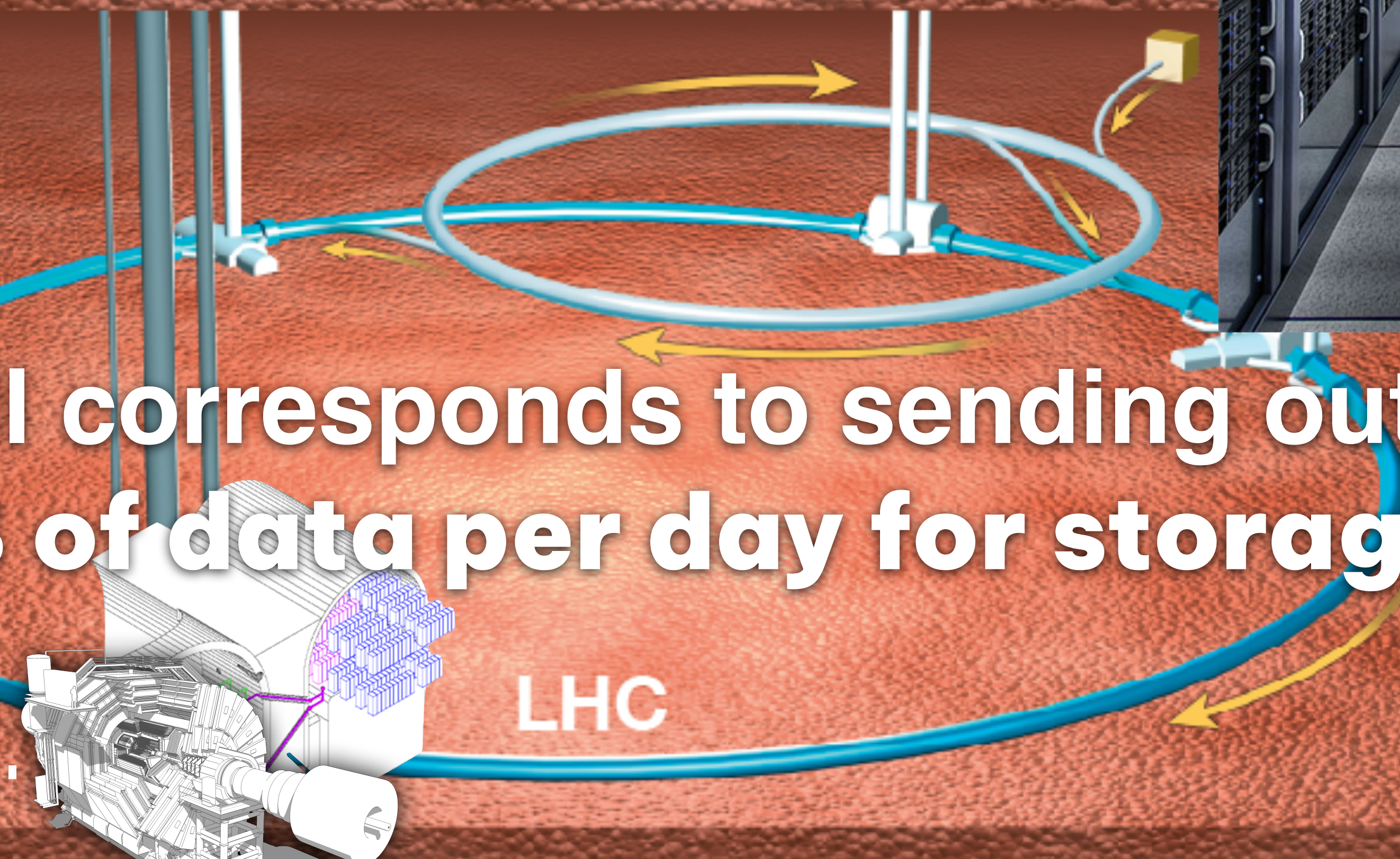
Every 6 months, would need:
1 million CPU years
3 Exabytes tape storage

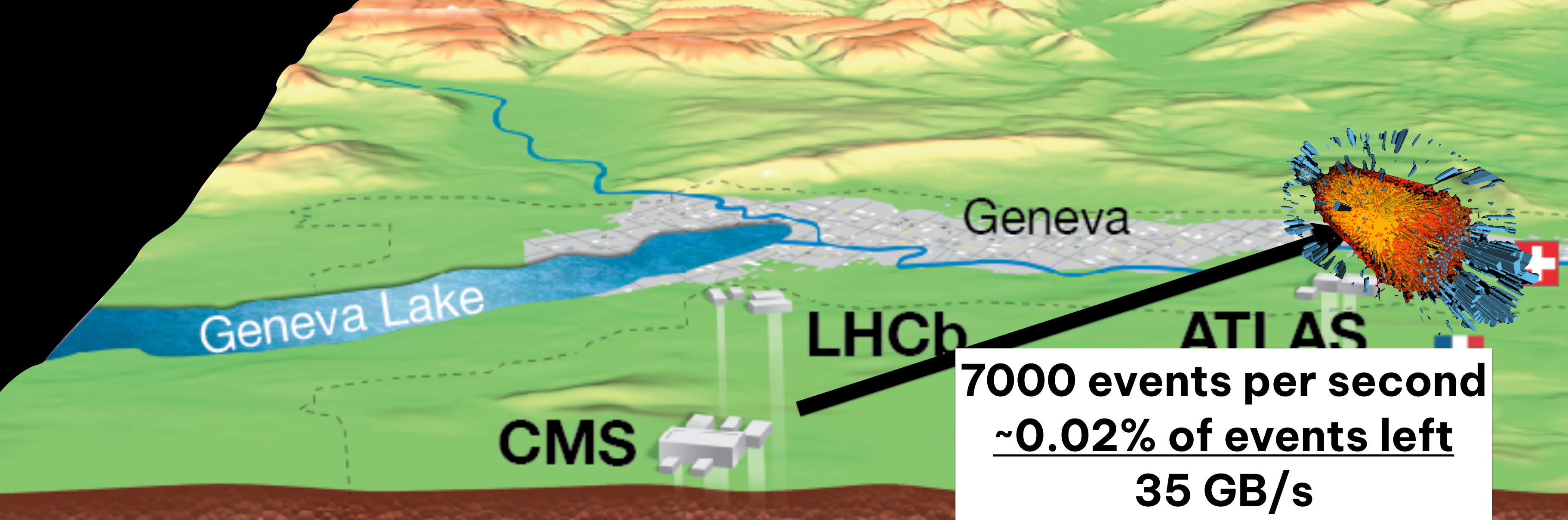
**Heterogeneous Software trigger:
25,600 CPUs and 400 GPUs
Reduce rate to 0.02%**



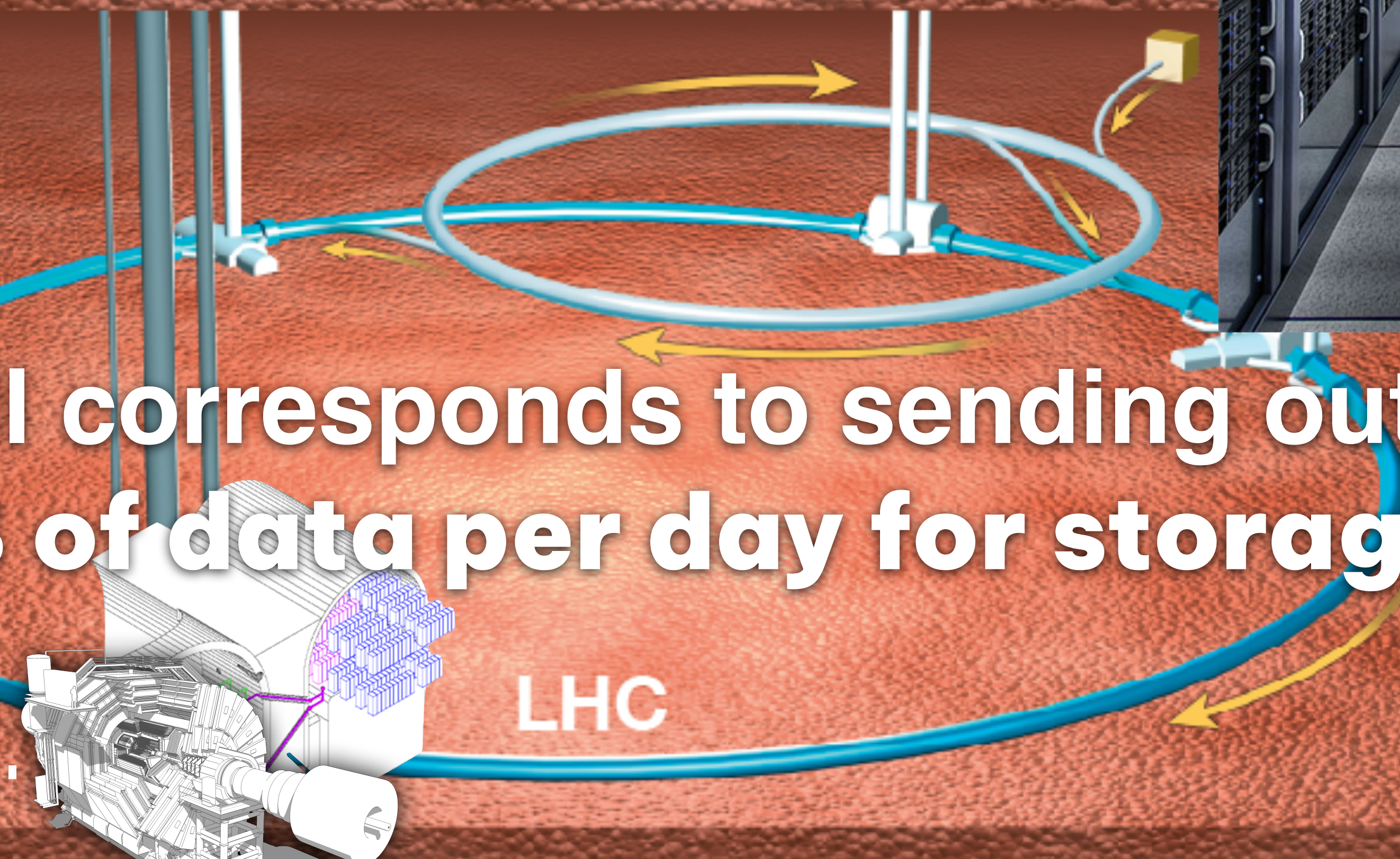


(Still corresponds to sending out
~a PB of data per day for storage)

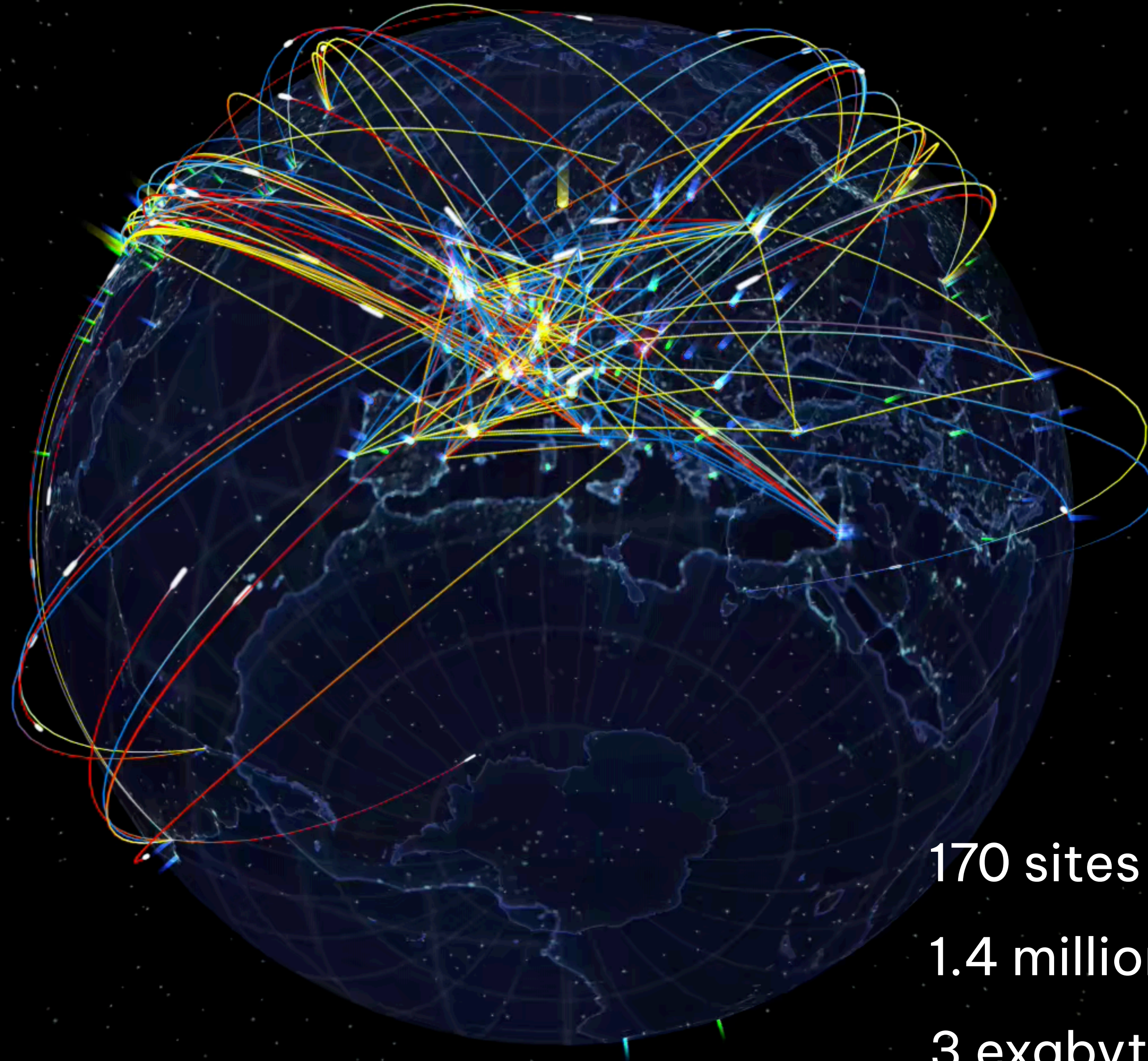




(Still corresponds to sending out
~a PB of data per day for storage)



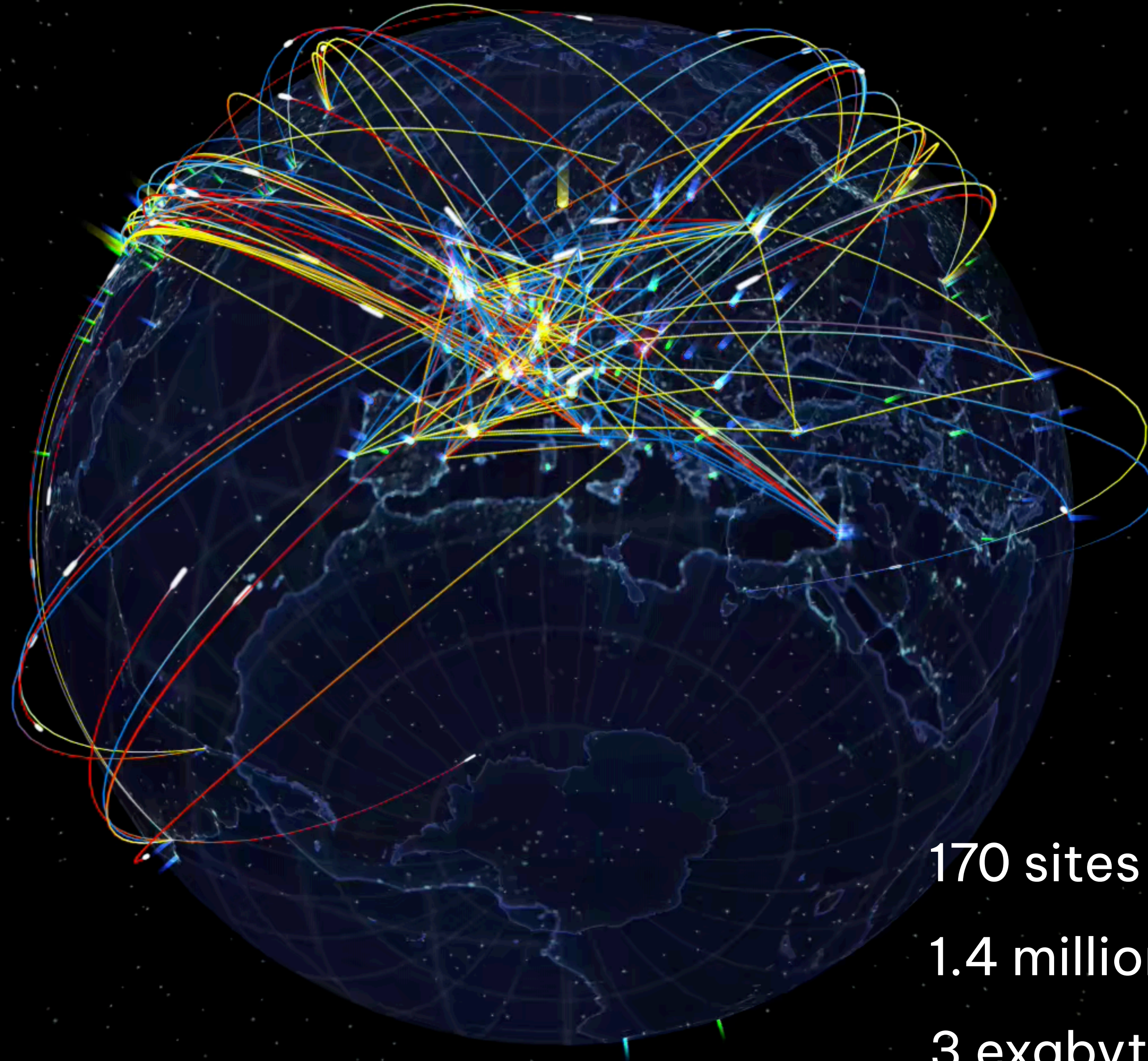
The Worldwide LHC Computing Grid



170 sites in 42 countries
1.4 million computer cores
3 exabytes of storage

Running jobs: 365644
Active CPU cores: 807139
Transfer rate: 21.54 GiB/sec

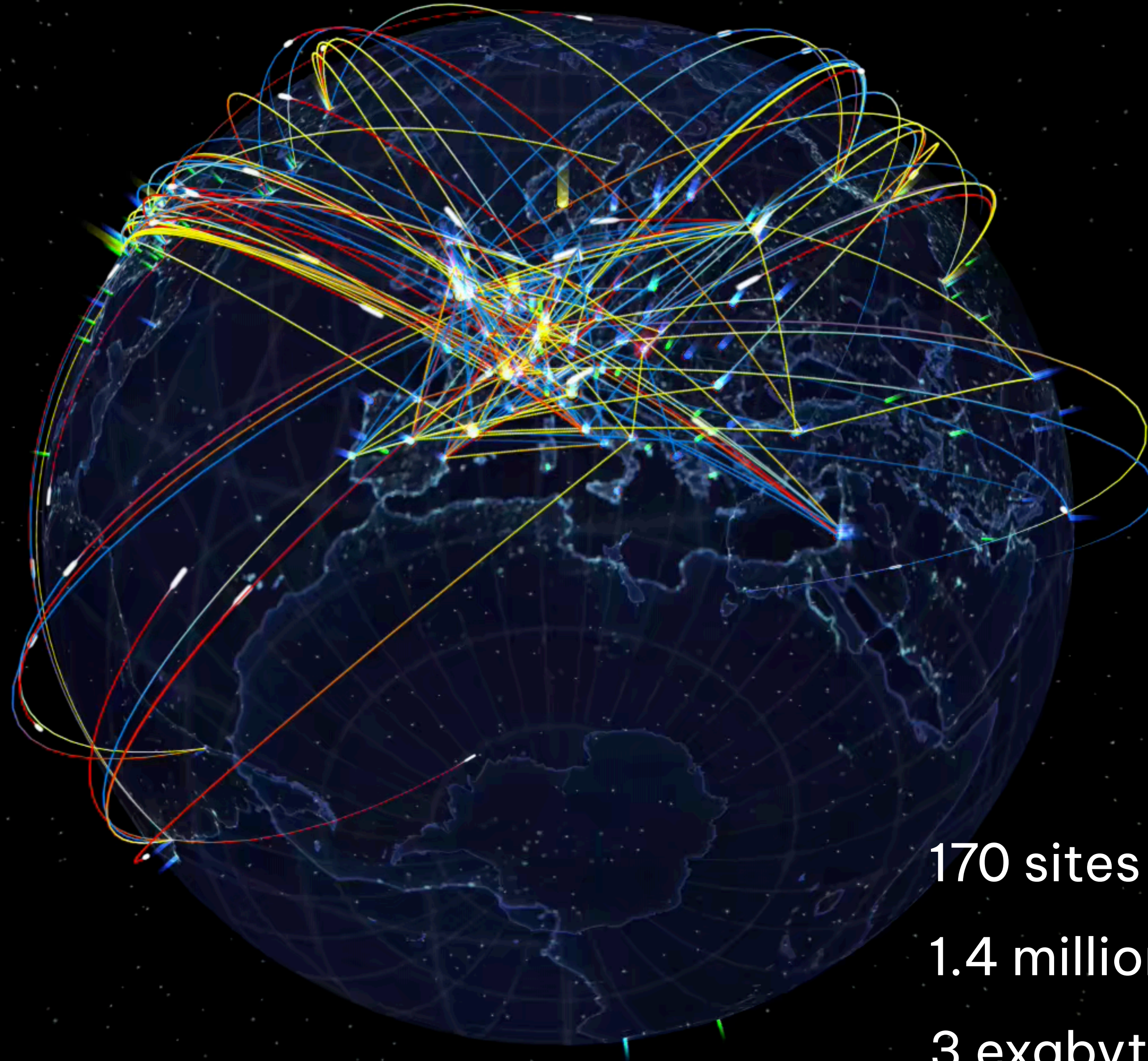
The Worldwide LHC Computing Grid



170 sites in 42 countries
1.4 million computer cores
3 exabytes of storage

Running jobs: 365644
Active CPU cores: 807139
Transfer rate: 21.54 GiB/sec

The Worldwide LHC Computing Grid



170 sites in 42 countries
1.4 million computer cores
3 exabytes of storage

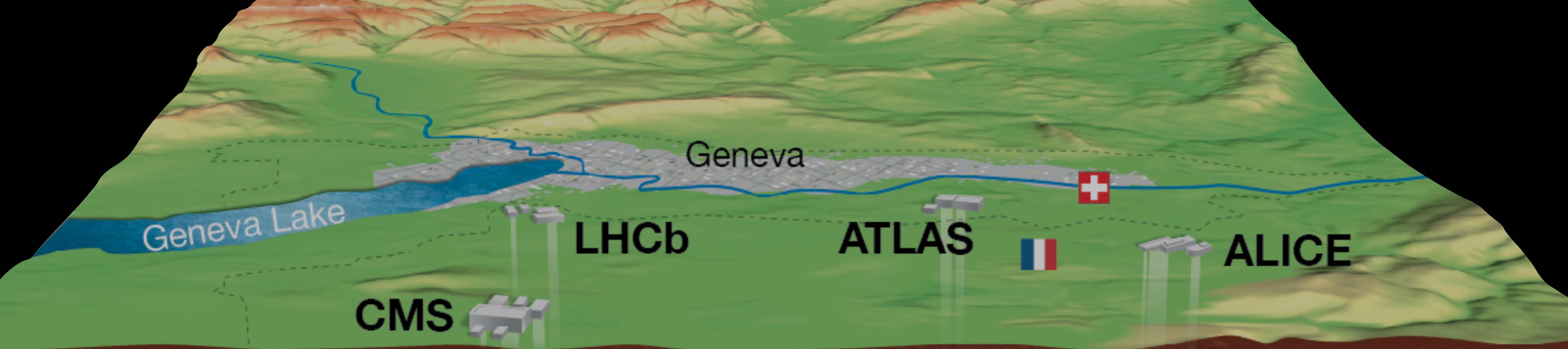
Running jobs: 365644
Active CPU cores: 807139
Transfer rate: 21.54 GiB/sec

The Standard Model

...and measuring its
19 free parameters
to 5σ (= 0.00003%
chance of statistical
fluctuation)...

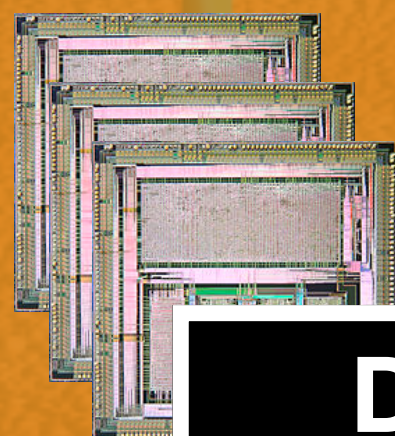
$$\begin{aligned}
 & -\frac{1}{2}\partial_\nu g_\mu^a \partial_\nu g_\mu^a - g_s f^{abc} \partial_\mu g_\nu^a g_\mu^b g_\nu^c - \frac{1}{4}g_s^2 f^{abc} f^{ade} g_\mu^b g_\nu^c g_\mu^d g_\nu^e + \\
 & \frac{1}{2}ig_s^2 (\bar{q}_i^\sigma \gamma^\mu q_j^\sigma) g_\mu^a + \bar{G}^a \partial^2 G^a + g_s f^{abc} \partial_\mu \bar{G}^a G^b g_\mu^c - \partial_\nu W_\mu^+ \partial_\nu W_\mu^- - \\
 & M^2 W_\mu^+ W_\mu^- - \frac{1}{2}\partial_\nu Z_\mu^0 \partial_\nu Z_\mu^0 - \frac{1}{2c_w^2} M^2 Z_\mu^0 Z_\mu^0 - \frac{1}{2}\partial_\mu A_\nu \partial_\mu A_\nu - \frac{1}{2}\partial_\mu H \partial_\mu H - \\
 & \frac{1}{2}m_h^2 H^2 - \partial_\mu \phi^+ \partial_\mu \phi^- - M^2 \phi^+ \phi^- - \frac{1}{2}\partial_\mu \phi^0 \partial_\mu \phi^0 - \frac{1}{2c_w^2} M \phi^0 \phi^0 - \beta_h [\frac{2M^2}{g^2} + \\
 & \frac{2M}{g} H + \frac{1}{2}(H^2 + \phi^0 \phi^0 + 2\phi^+ \phi^-)] + \frac{2M^4}{g^2} \alpha_h - igc_w [\partial_\nu Z_\mu^0 (W_\mu^+ W_\nu^- - \\
 & W_\nu^+ W_\mu^-) - Z_\nu^0 (W_\mu^+ \partial_\nu W_\mu^- - W_\mu^- \partial_\nu W_\mu^+) + Z_\mu^0 (W_\nu^+ \partial_\nu W_\mu^- - \\
 & W_\nu^- \partial_\nu W_\mu^+)] - igs_w [\partial_\nu A_\mu (W_\mu^+ W_\nu^- - W_\nu^+ W_\mu^-) - A_\nu (W_\mu^+ \partial_\nu W_\mu^- - \\
 & W_\mu^- \partial_\nu W_\mu^+) + A_\mu (W_\nu^+ \partial_\nu W_\mu^- - W_\nu^- \partial_\nu W_\mu^+)] - \frac{1}{2}g^2 W_\mu^+ W_\mu^- W_\nu^+ W_\nu^- + \\
 & \frac{1}{2}g^2 W_\mu^+ W_\nu^- W_\mu^+ W_\nu^- + g^2 c_w^2 (Z_\mu^0 W_\mu^+ Z_\nu^0 W_\nu^- - Z_\mu^0 Z_\nu^0 W_\mu^+ W_\nu^-) + \\
 & g^2 s_w^2 (A_\mu W_\mu^+ A_\nu W_\nu^- - A_\mu A_\nu W_\mu^+ W_\nu^-) + g^2 s_w c_w [A_\mu Z_\nu^0 (W_\mu^+ W_\nu^- - \\
 & W_\nu^+ W_\mu^-) - 2A_\mu Z_\mu^0 W_\nu^+ W_\nu^-] - g\alpha [H^3 + H\phi^0 \phi^0 + 2H\phi^+ \phi^-] - \\
 & \frac{1}{8}g^2 \alpha_h [H^4 + (\phi^0)^4 + 4(\phi^+ \phi^-)^2 + 4(\phi^0)^2 \phi^+ \phi^- + 4H^2 \phi^+ \phi^- + 2(\phi^0)^2 H^2] - \\
 & gMW_\mu^+ W_\mu^- H - \frac{1}{2}g\frac{M}{c_w^2} Z_\mu^0 Z_\mu^0 H - \frac{1}{2}ig[W_\mu^+ (\phi^0 \partial_\mu \phi^- - \phi^- \partial_\mu \phi^0) - \\
 & W_\mu^- (\phi^0 \partial_\mu \phi^+ - \phi^+ \partial_\mu \phi^0)] + \frac{1}{2}g[W_\mu^+ (H \partial_\mu \phi^- - \phi^- \partial_\mu H) - W_\mu^- (H \partial_\mu \phi^+ - \\
 & \phi^+ \partial_\mu H)] + \frac{1}{2}g\frac{1}{c_w} (Z_\mu^0 (H \partial_\mu \phi^0 - \phi^0 \partial_\mu H) - ig\frac{s_w^2}{c_w} M Z_\mu^0 (W_\mu^+ \phi^- - W_\mu^- \phi^+) + \\
 & igs_w M A_\mu (W_\mu^+ \phi^- - W_\mu^- \phi^+) - ig\frac{1-2c_w^2}{2c_w} Z_\mu^0 (\phi^+ \partial_\mu \phi^- - \phi^- \partial_\mu \phi^+) + \\
 & igs_w A_\mu (\phi^+ \partial_\mu \phi^- - \phi^- \partial_\mu \phi^+) - \frac{1}{4}g^2 W_\mu^+ W_\mu^- [H^2 + (\phi^0)^2 + 2\phi^+ \phi^-] - \\
 & \frac{1}{4}g^2 \frac{1}{c_w^2} Z_\mu^0 Z_\mu^0 [H^2 + (\phi^0)^2 + 2(2s_w^2 - 1)^2 \phi^+ \phi^-] - \frac{1}{2}g^2 \frac{s_w^2}{c_w} Z_\mu^0 \phi^0 (W_\mu^+ \phi^- + \\
 & W_\mu^- \phi^+) - \frac{1}{2}ig^2 \frac{s_w^2}{c_w} Z_\mu^0 H (W_\mu^+ \phi^- - W_\mu^- \phi^+) + \frac{1}{2}g^2 s_w A_\mu \phi^0 (W_\mu^+ \phi^- + \\
 & W_\mu^- \phi^+) + \frac{1}{2}ig^2 s_w A_\mu H (W_\mu^+ \phi^- - W_\mu^- \phi^+) - g^2 \frac{s_w}{c_w} (2c_w^2 - 1) Z_\mu^0 A_\mu \phi^+ \phi^- - \\
 & g^1 s_w^2 A_\mu A_\mu \phi^+ \phi^- - \bar{e}^\lambda (\gamma^\partial + m_e^\lambda) e^\lambda - \bar{\nu}^\lambda \gamma^\partial \nu^\lambda - \bar{u}_j^\lambda (\gamma^\partial + m_u^\lambda) u_j^\lambda - \\
 & \bar{d}_j^\lambda (\gamma^\partial + m_d^\lambda) d_j^\lambda + igs_w A_\mu [-(\bar{e}^\lambda \gamma^\mu e^\lambda) + \frac{2}{3}(\bar{u}_j^\lambda \gamma^\mu u_j^\lambda) - \frac{1}{3}(\bar{d}_j^\lambda \gamma^\mu d_j^\lambda)] + \\
 & \frac{ig}{4c_w} Z_\mu^0 [(\bar{\nu}^\lambda \gamma^\mu (1 + \gamma^5) \nu^\lambda) + (\bar{e}^\lambda \gamma^\mu (4s_w^2 - 1 - \gamma^5) e^\lambda) + (\bar{u}_j^\lambda \gamma^\mu (\frac{4}{3}s_w^2 - \\
 & 1 - \gamma^5) u_j^\lambda) + (\bar{d}_j^\lambda \gamma^\mu (1 - \frac{8}{3}s_w^2 - \gamma^5) d_j^\lambda)] + \frac{ig}{2\sqrt{2}} W_\mu^+ [(\bar{\nu}^\lambda \gamma^\mu (1 + \gamma^5) \nu^\lambda) + \\
 & (\bar{u}_j^\lambda \gamma^\mu (1 + \gamma^5) C_{\lambda\kappa} d_j^\kappa)] + \frac{ig}{2\sqrt{2}} W_\mu^- [(\bar{e}^\lambda \gamma^\mu (1 + \gamma^5) \nu^\lambda) + (\bar{d}_j^\kappa C_{\lambda\kappa}^\dagger \gamma^\mu (1 + \\
 & \gamma^5) u_j^\lambda)] + \frac{ig}{2\sqrt{2}} \frac{m_e^\lambda}{M} [-\phi^+ (\bar{\nu}^\lambda (1 - \gamma^5) e^\lambda) + \phi^- (\bar{e}^\lambda (1 + \gamma^5) \nu^\lambda)] - \\
 & \frac{g}{2} \frac{m_e^\lambda}{M} [H(\bar{e}^\lambda e^\lambda) + i\phi^0 (\bar{e}^\lambda \gamma^5 e^\lambda)] + \frac{ig}{2M\sqrt{2}} \phi^+ [-m_d^\kappa (\bar{u}_j^\lambda C_{\lambda\kappa} (1 - \gamma^5) d_j^\kappa) + \\
 & m_u^\lambda (\bar{u}_j^\lambda C_{\lambda\kappa} (1 + \gamma^5) d_j^\kappa) + \frac{ig}{2M\sqrt{2}} \phi^- [m_d^\lambda (\bar{d}_j^\lambda C_{\lambda\kappa}^\dagger (1 + \gamma^5) u_j^\kappa) - m_u^\kappa (\bar{d}_j^\lambda C_{\lambda\kappa}^\dagger (1 - \\
 & \gamma^5) u_j^\kappa) - \frac{g}{2} \frac{m_d^\lambda}{M} H(\bar{u}_j^\lambda u_j^\lambda) - \frac{g}{2} \frac{m_d^\lambda}{M} H(\bar{d}_j^\lambda d_j^\lambda) + \frac{ig}{2} \frac{m_d^\lambda}{M} \phi^0 (\bar{u}_j^\lambda \gamma^5 u_j^\lambda) - \\
 & \frac{ig}{2} \frac{m_d^\lambda}{M} \phi^0 (\bar{d}_j^\lambda \gamma^5 d_j^\lambda) + \bar{X}^+ (\partial^2 - M^2) X^+ + \bar{X}^- (\partial^2 - M^2) X^- + \bar{X}^0 (\partial^2 - \\
 & \frac{M^2}{c_w^2}) X^0 + \bar{Y} \partial^2 Y + igc_w W_\mu^+ (\partial_\mu \bar{X}^0 X^- - \partial_\mu \bar{X}^+ X^0) + igs_w W_\mu^+ (\partial_\mu \bar{Y} X^- - \\
 & \partial_\mu \bar{X}^+ Y) + igc_w W_\mu^- (\partial_\mu \bar{X}^- X^0 - \partial_\mu \bar{X}^0 X^+) + igs_w W_\mu^- (\partial_\mu \bar{X}^- Y - \\
 & \partial_\mu \bar{Y} X^+) + igc_w Z_\mu^0 (\partial_\mu \bar{X}^+ X^- - \partial_\mu \bar{X}^- X^+) + igs_w A_\mu (\partial_\mu \bar{X}^+ X^- - \\
 & \partial_\mu \bar{X}^- X^+) - \frac{1}{2}gM [\bar{X}^+ X^+ H + \bar{X}^- X^- H + \frac{1}{c_w^2} \bar{X}^0 X^0 H] + \\
 & \frac{1-2c_w^2}{2c_w} igM [\bar{X}^+ X^0 \phi^+ - \bar{X}^- X^0 \phi^-] + \frac{1}{2c_w} igM [\bar{X}^0 X^- \phi^+ - \bar{X}^0 X^+ \phi^-] + \\
 & igM s_w [\bar{X}^0 X^- \phi^+ - \bar{X}^0 X^+ \phi^-] + \frac{1}{2}igM [\bar{X}^+ X^+ \phi^0 - \bar{X}^- X^- \phi^0]
 \end{aligned}$$

**Real-time Machine Learning essential for
extracting the best 0.02% for extreme science**



How did we get here?

ASIC inference



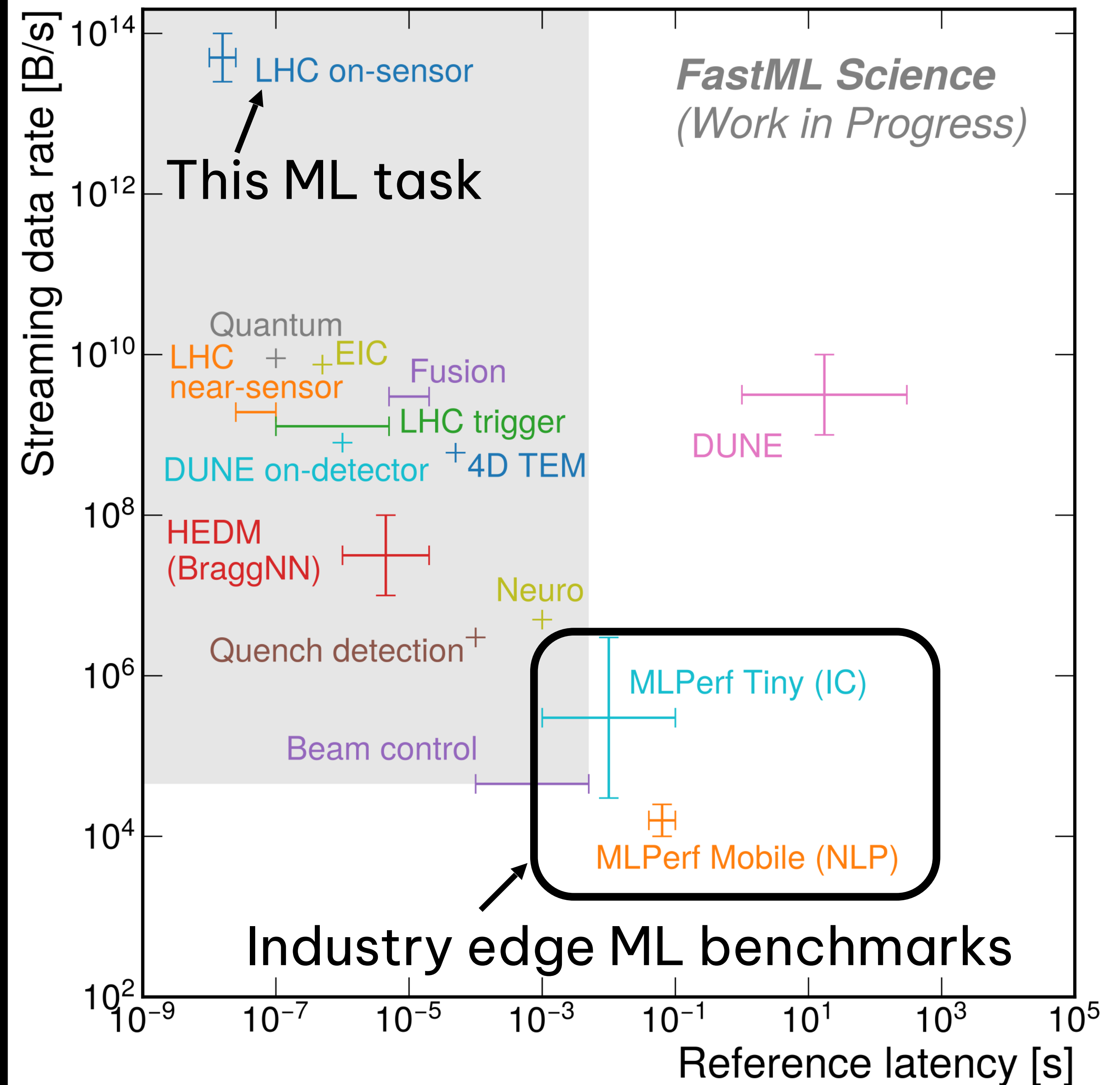
**Detector:
Latency $O(1)$ ns**

FPGA inference

**Level-1 trigger:
Latency $O(10)$ ns**



***Had to design our own tools
(to do inference in <100 ns)***



Floating point:

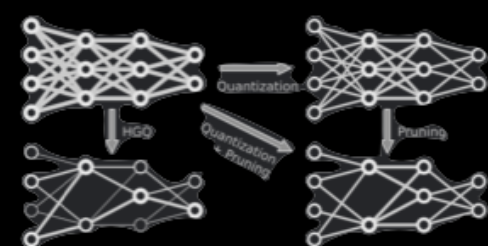


Model

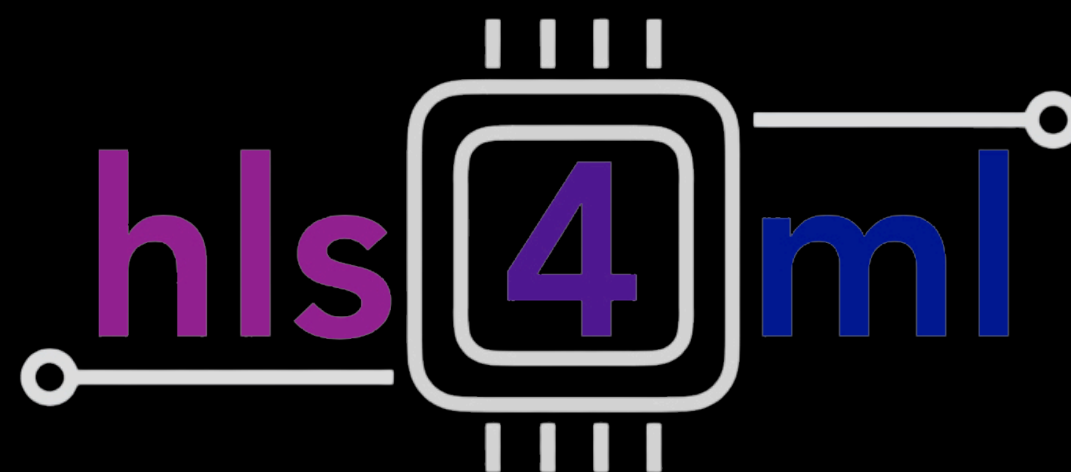
Quantized:



Brevitas

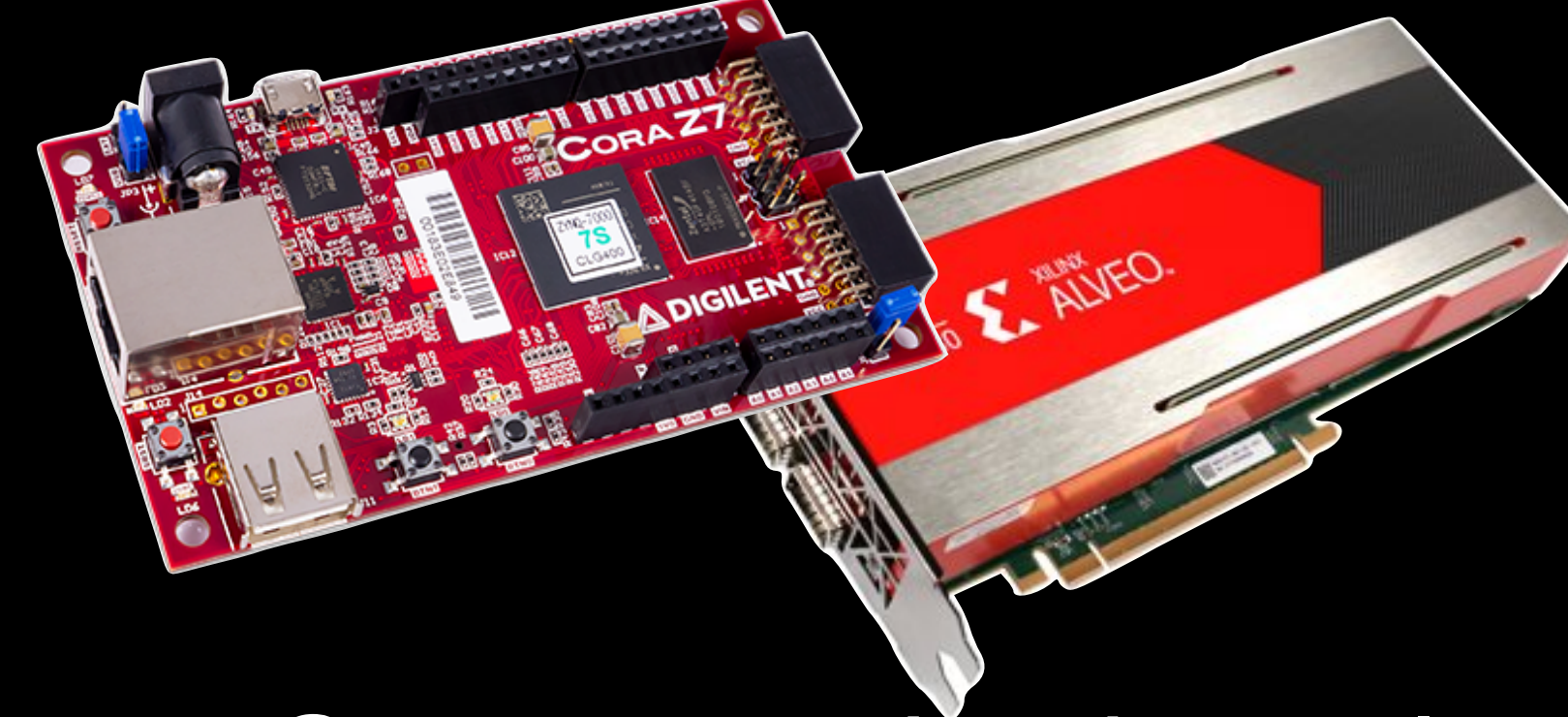


HQG



HLS Model

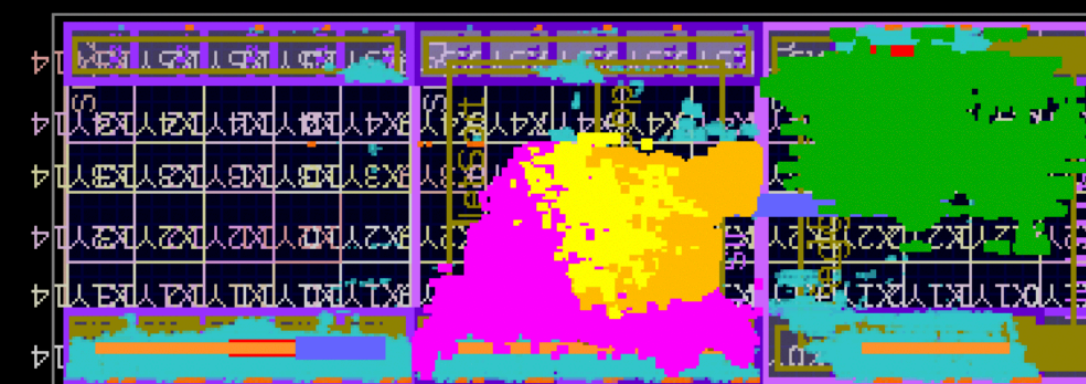
Write HLS project
Emulate
Run synthesis



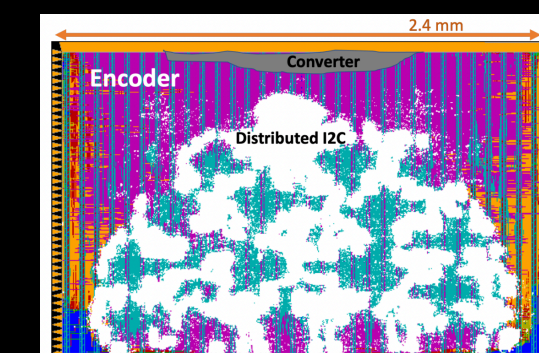
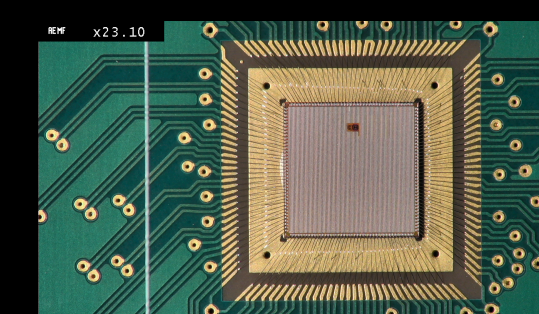
Co-processing kernel
(Accelerators/SoCs)

Hardware
Model

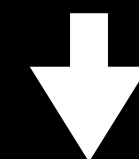
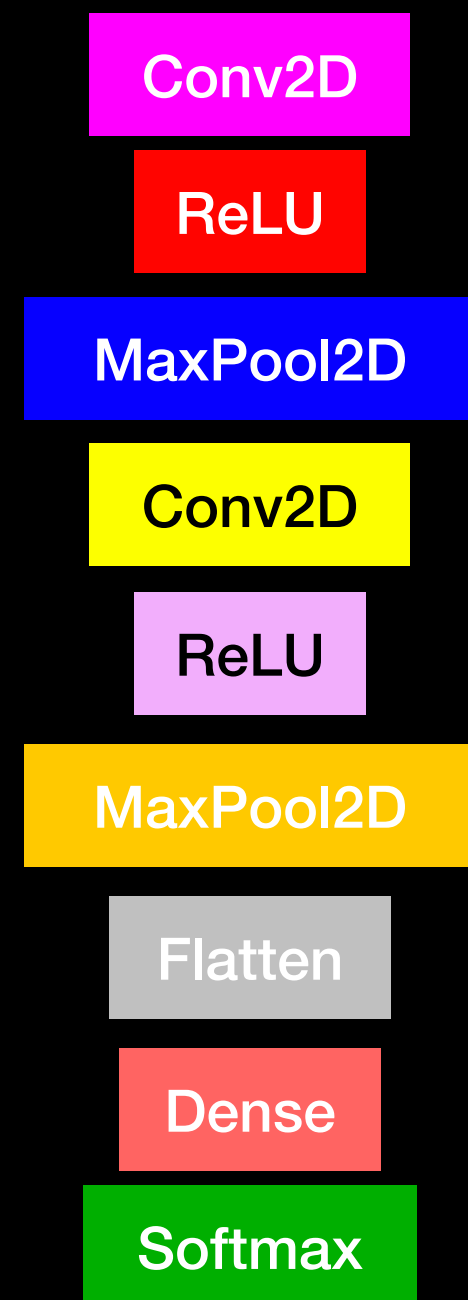
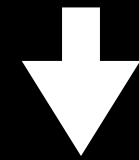
FPGA custom designs



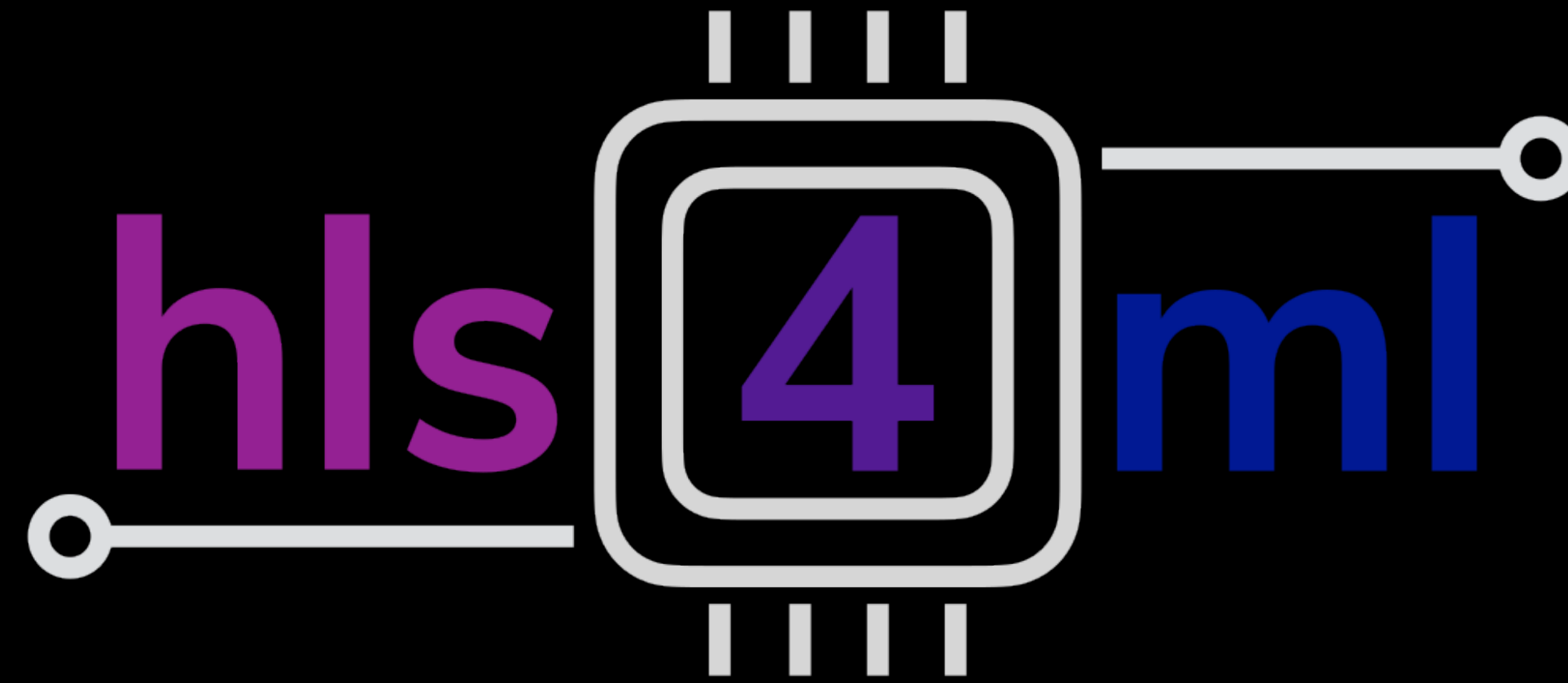
ASICs



0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9

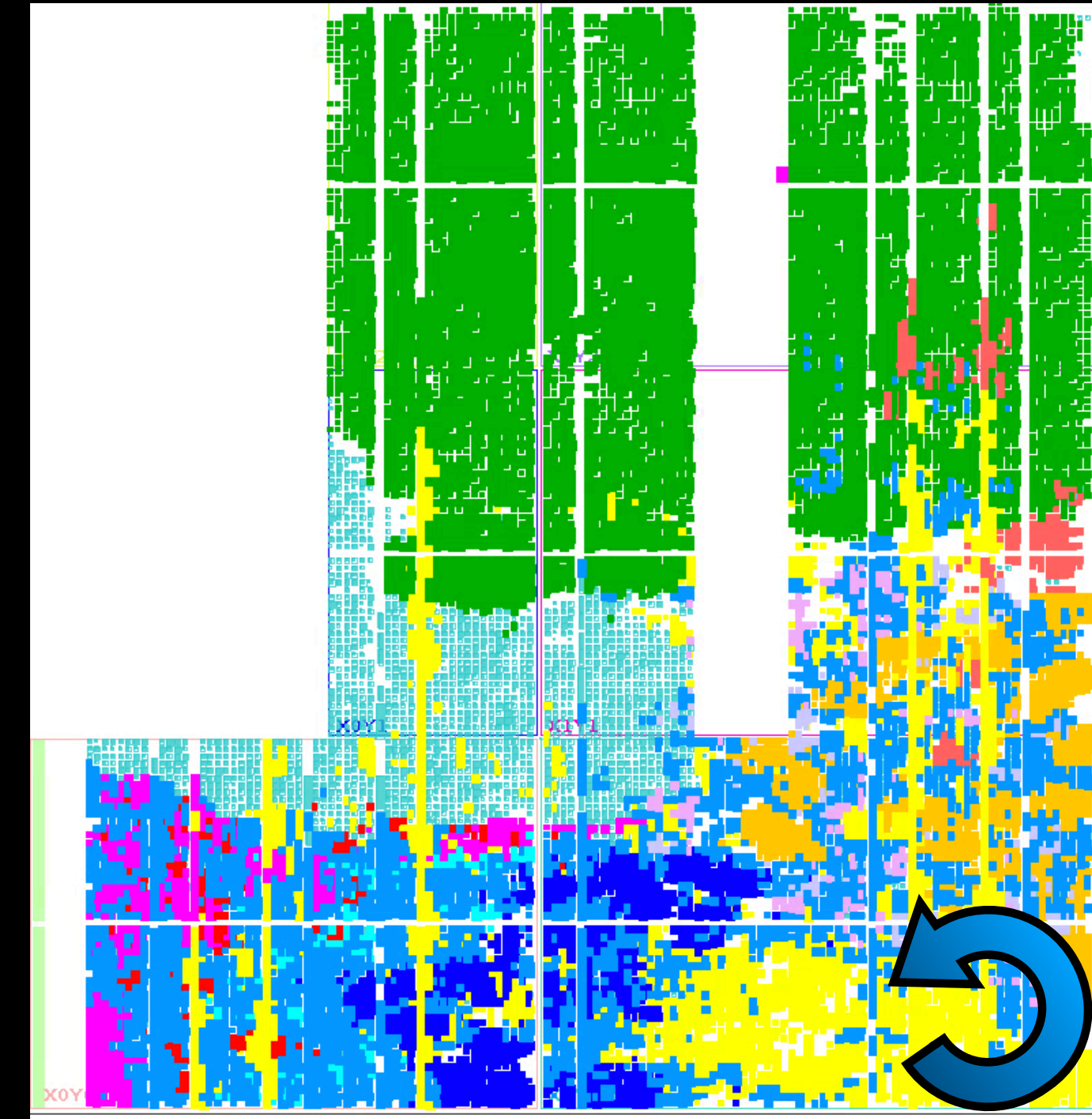


Prediction

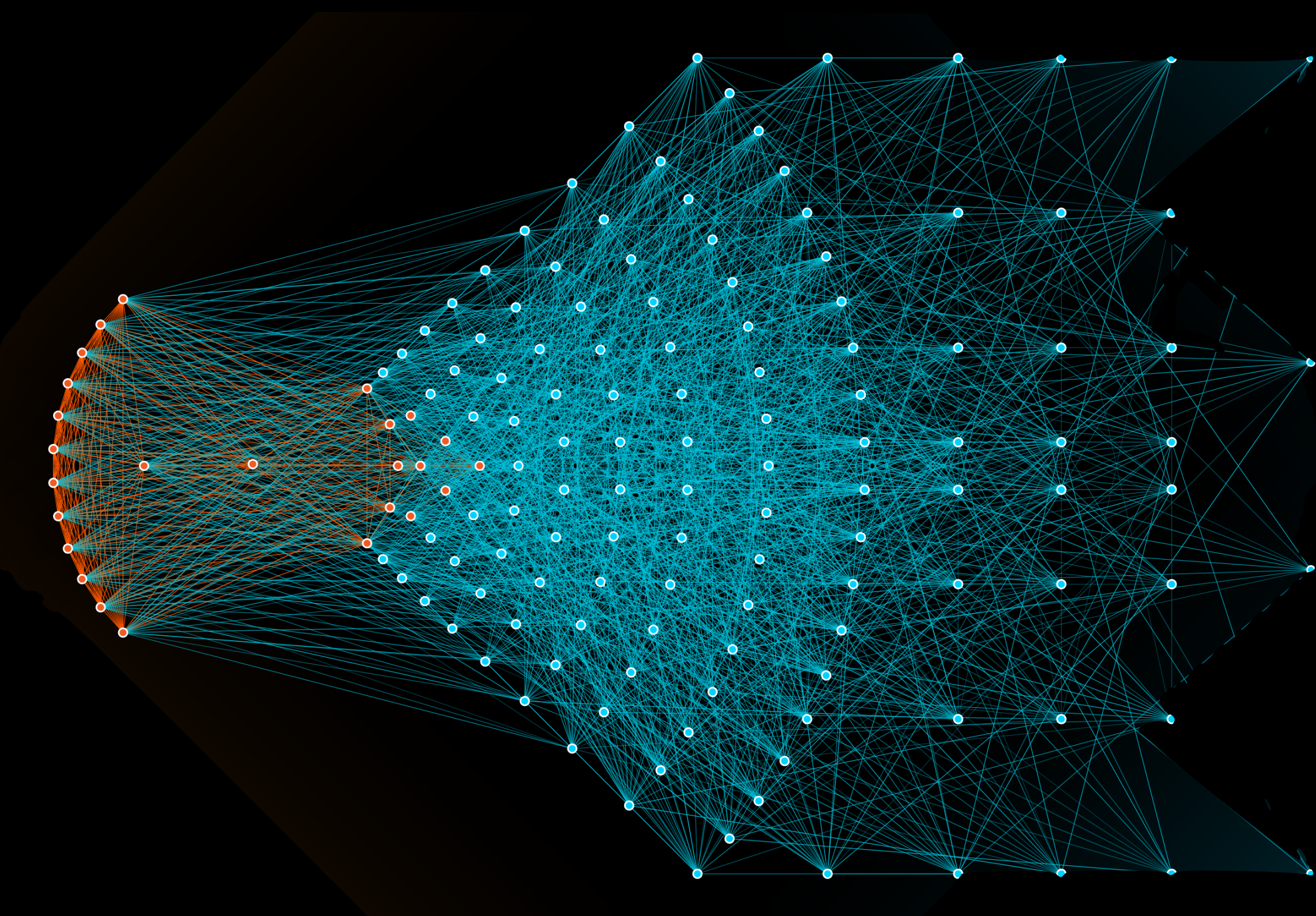


Data flow architecture

- Tailored hardware for a model (“Decisions are design time”)
- Each layer is separate compute unit
- Stay on-chip



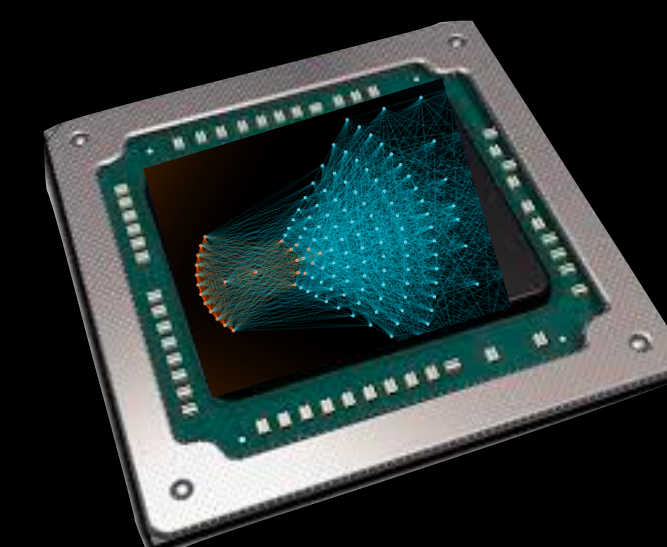
Dataflow



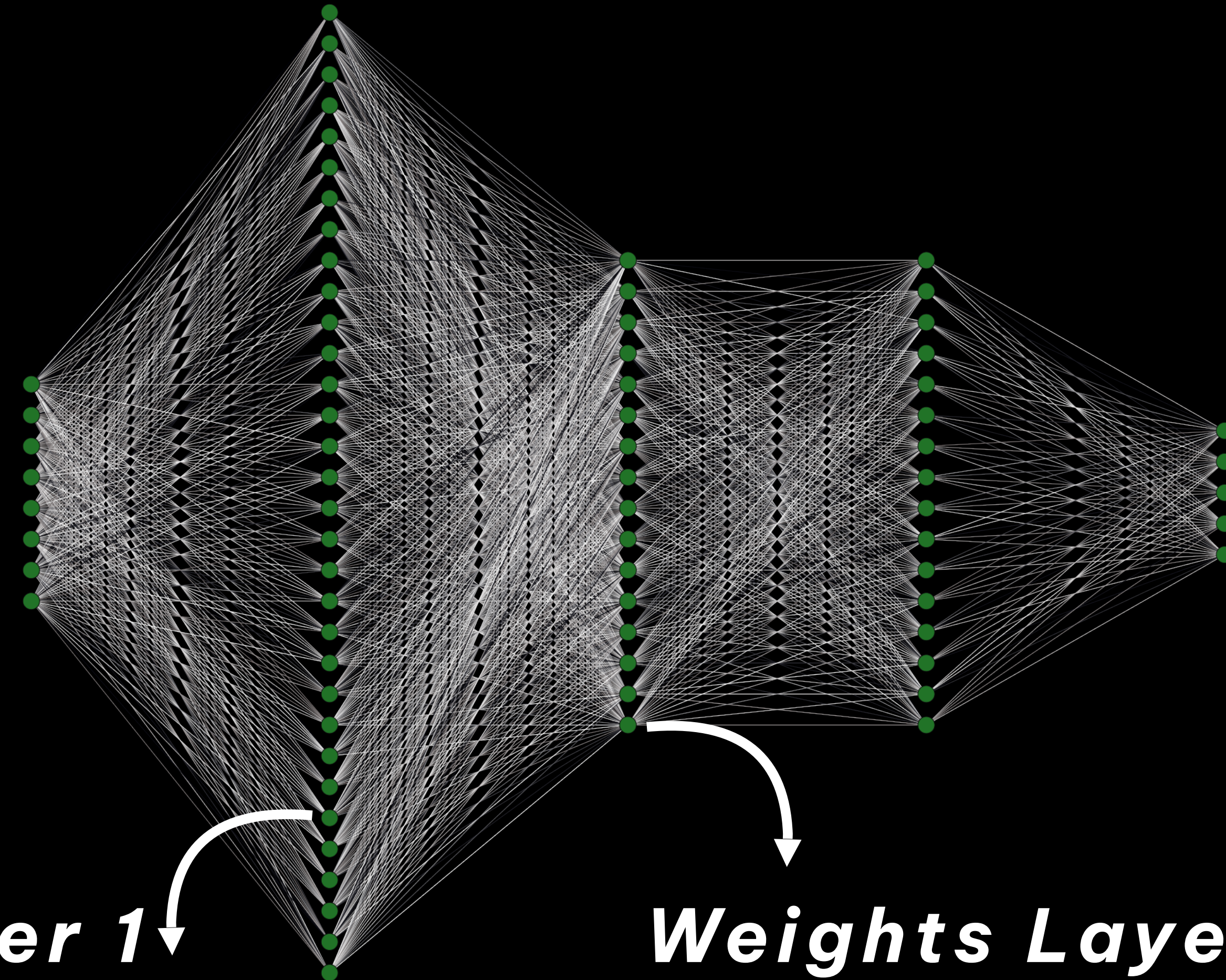
Ideally



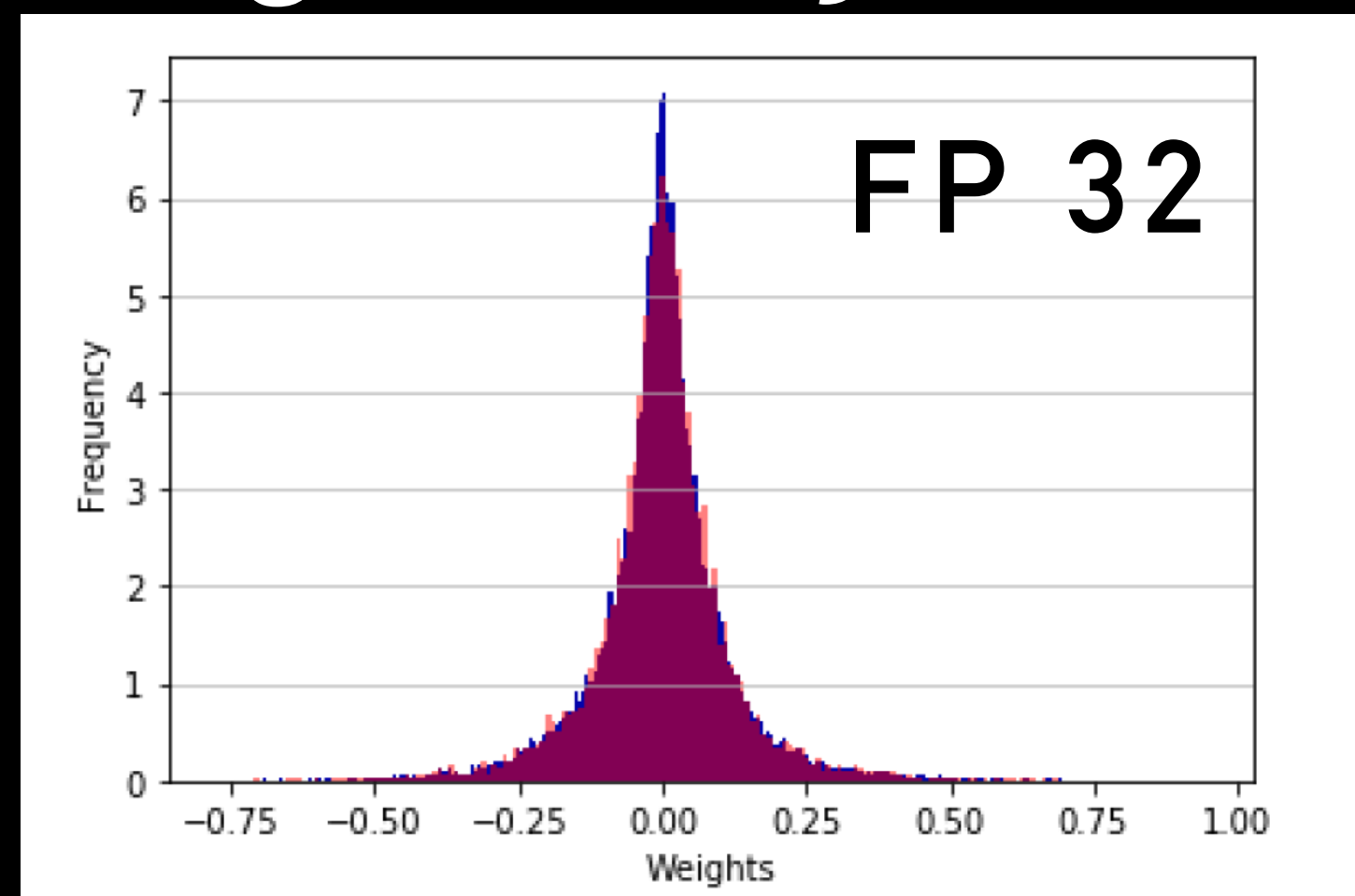
- Quantization
- Pruning
- Parallelisation
- Knowledge distillation



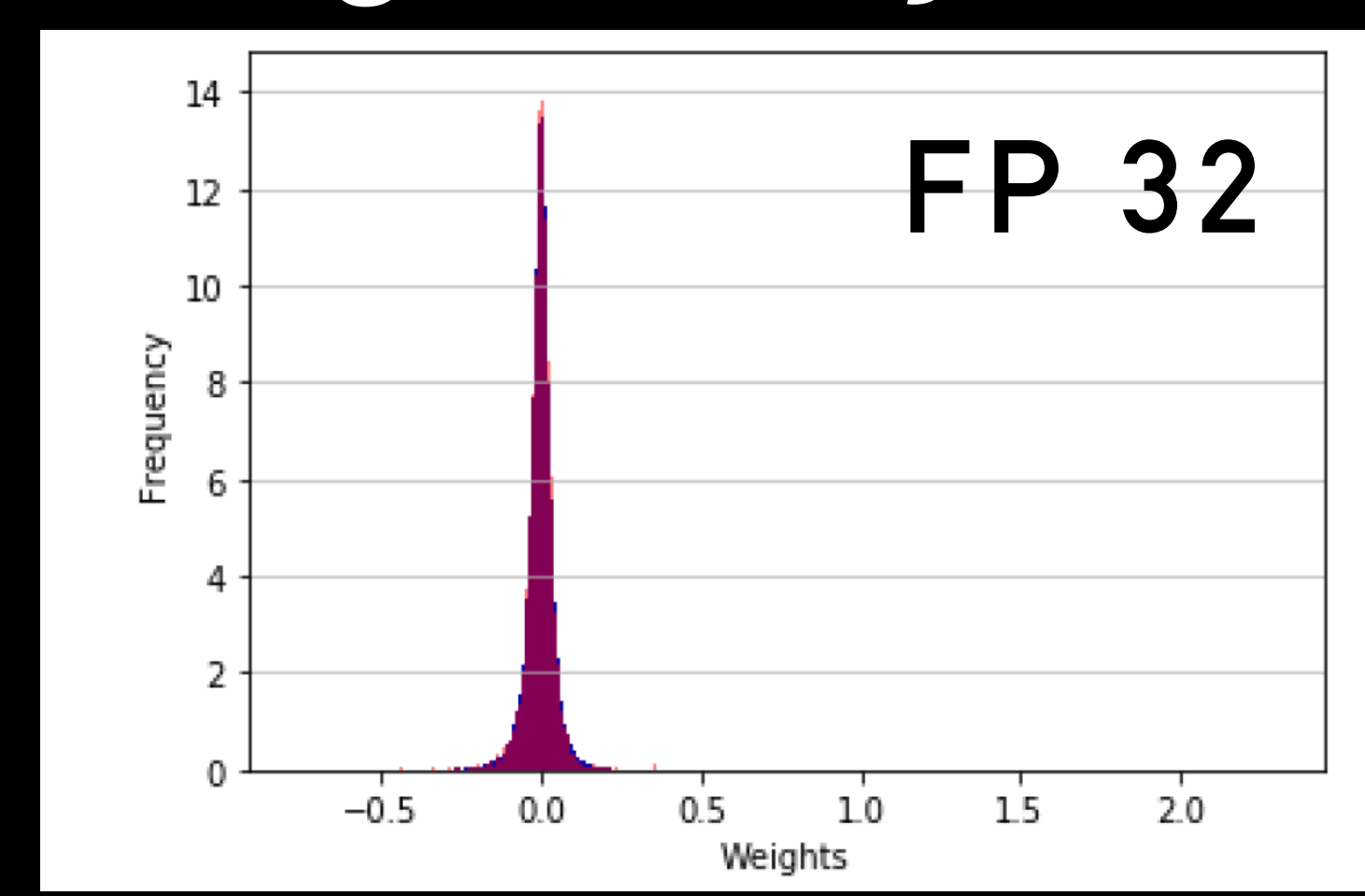
Reality



Weights Layer 1

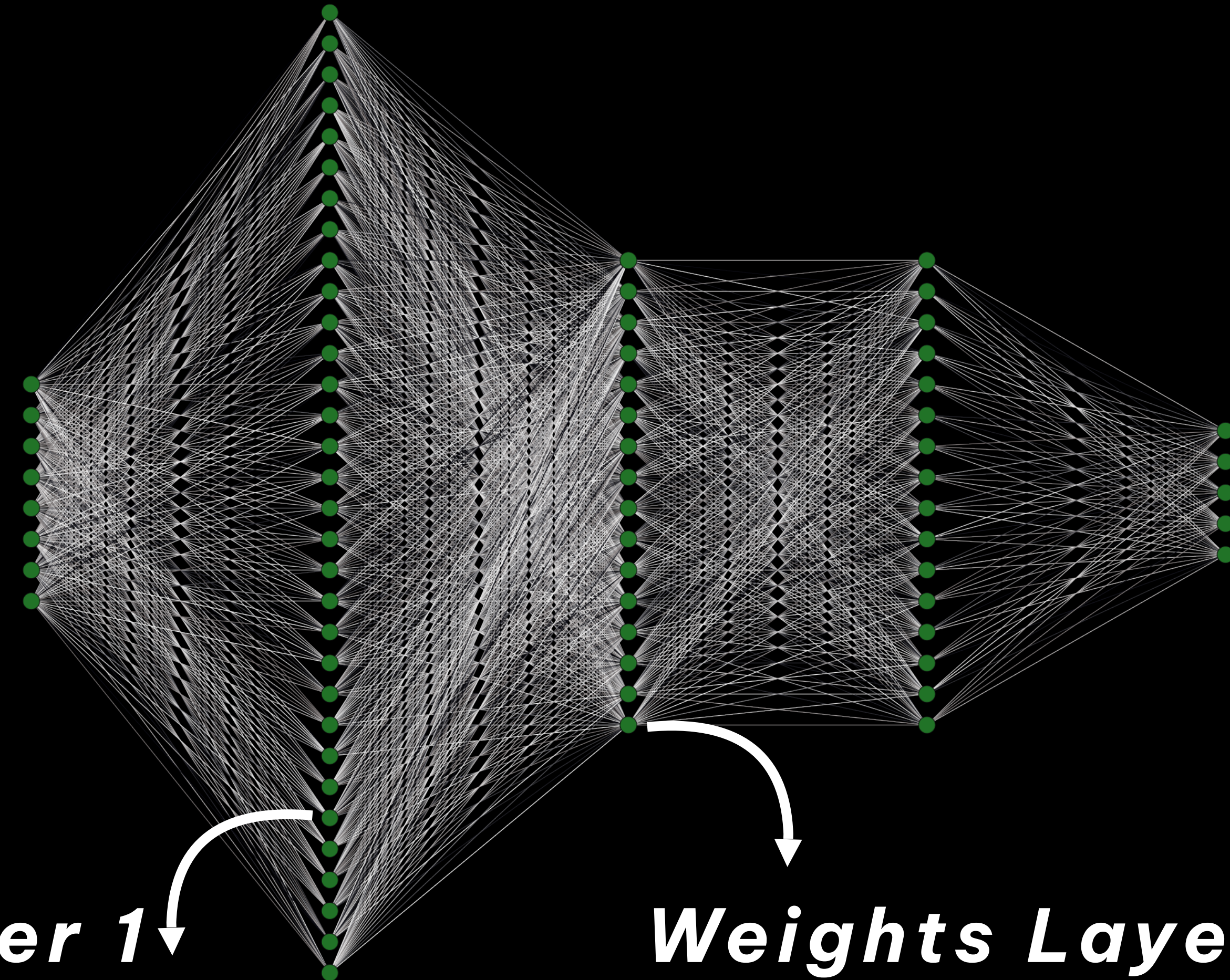


Weights Layer 2

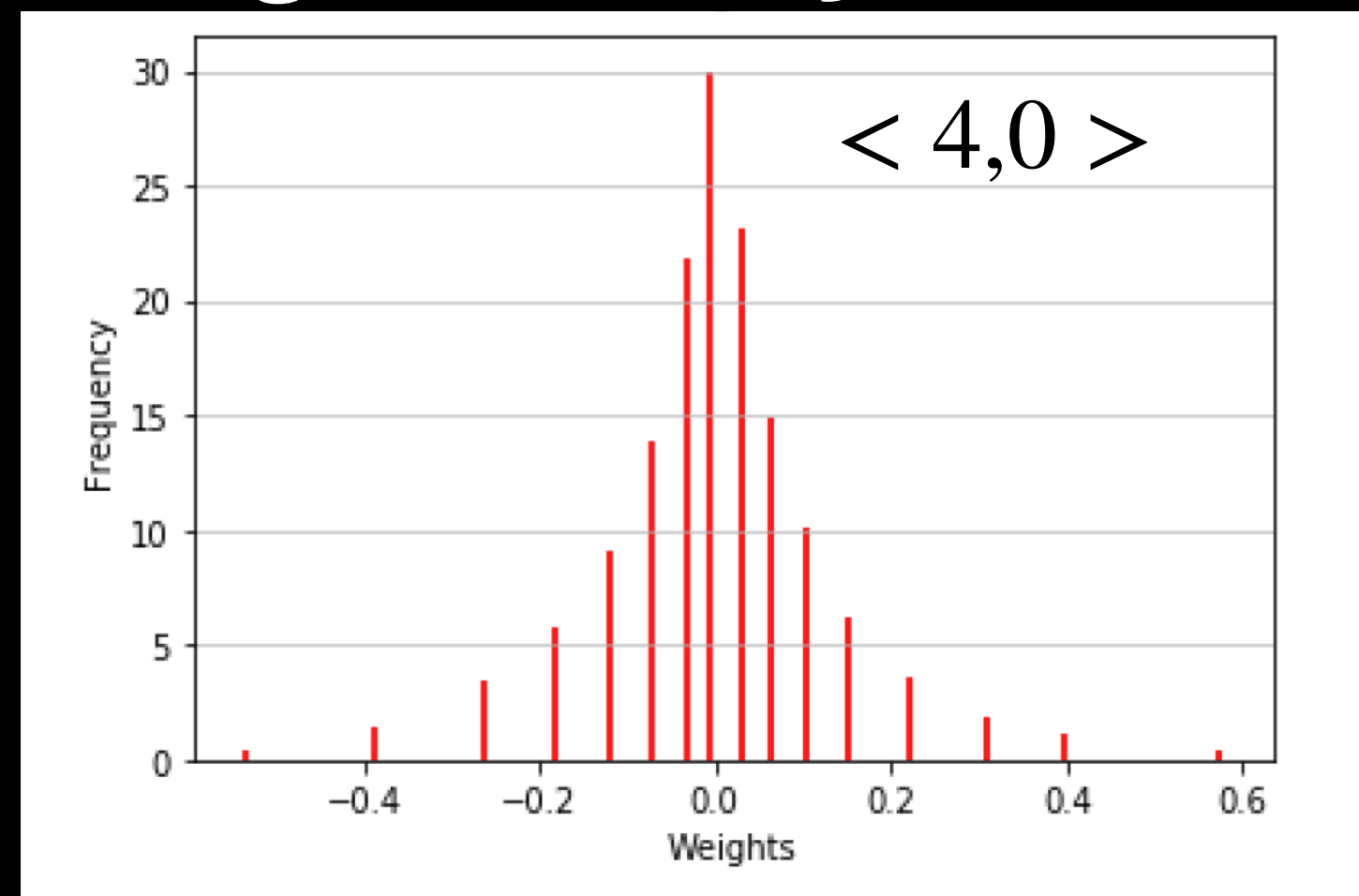


Fixed point

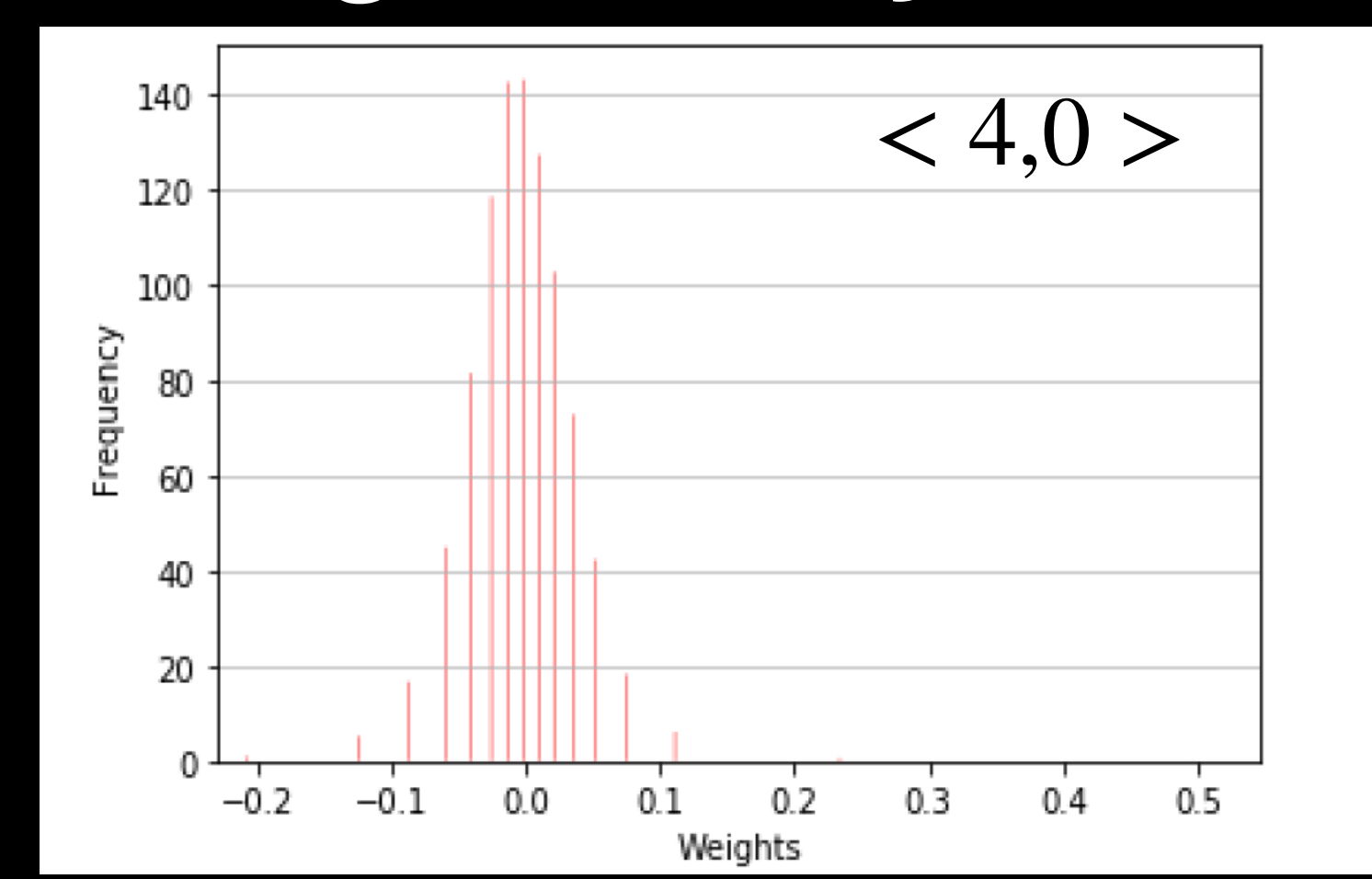
0101.1011101010

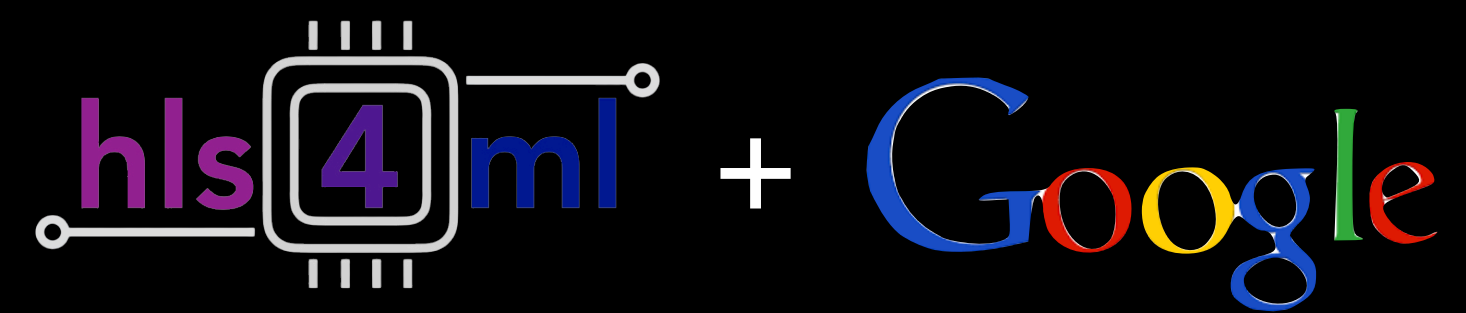


Weights Layer 1

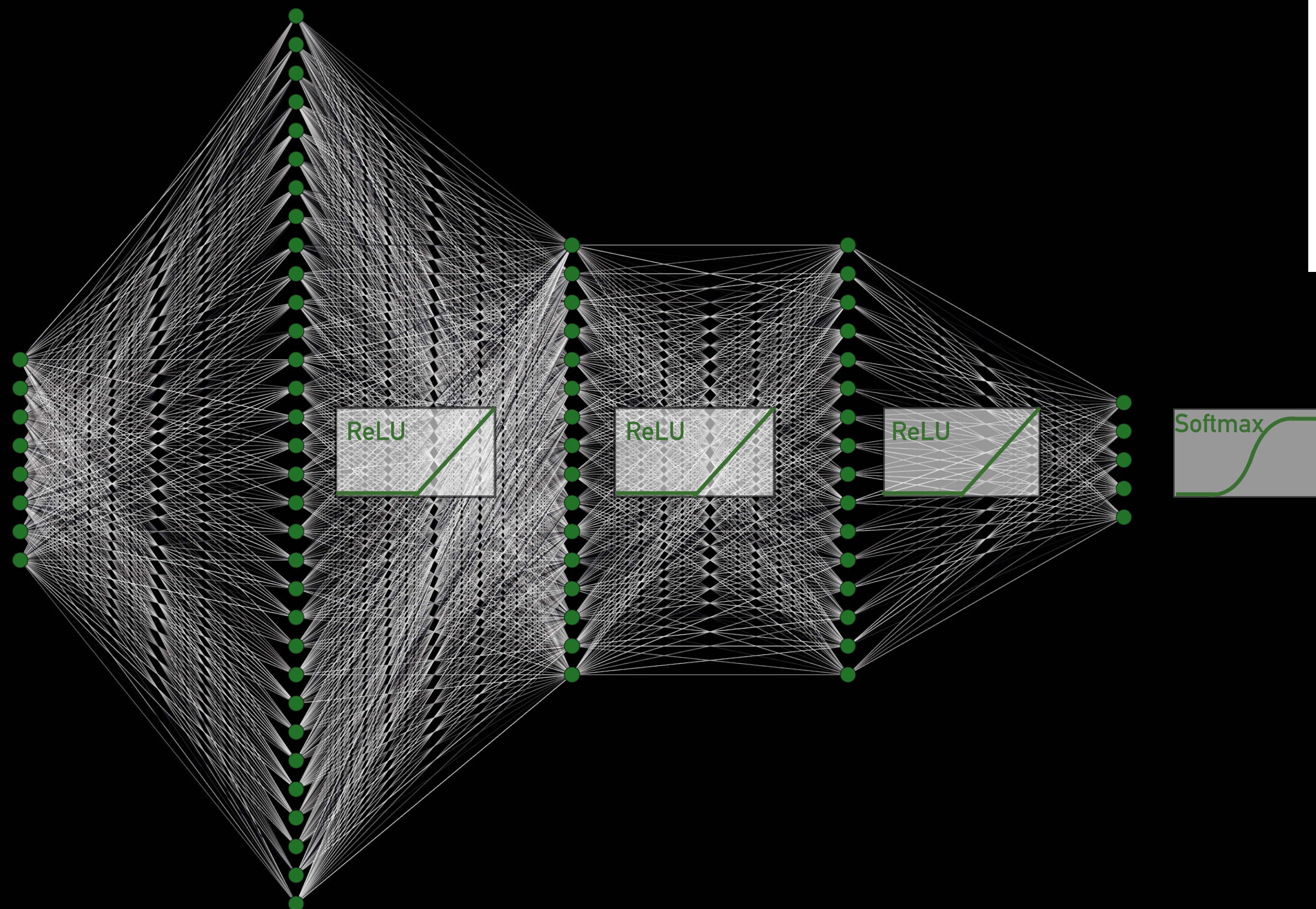


Weights Layer 2

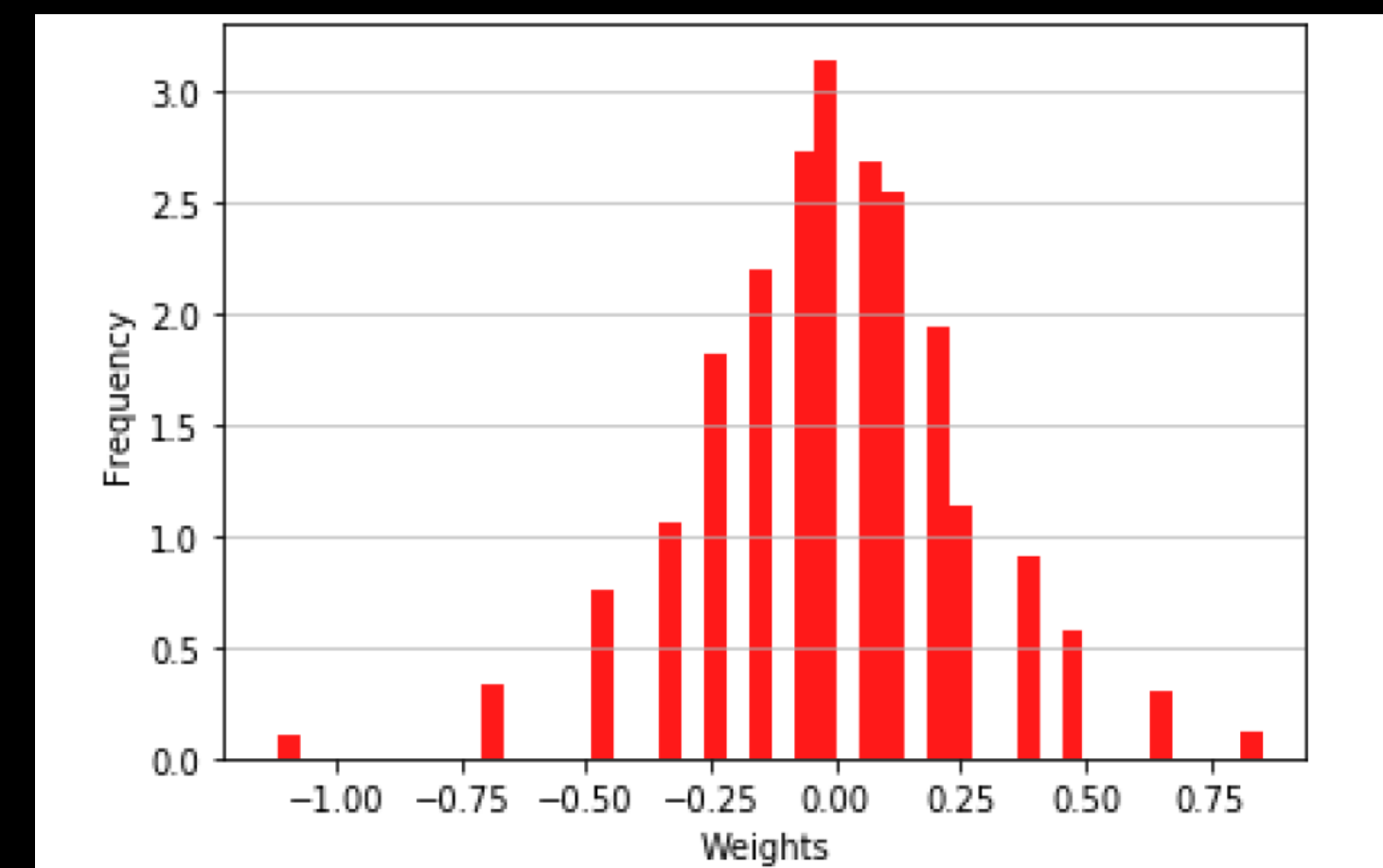




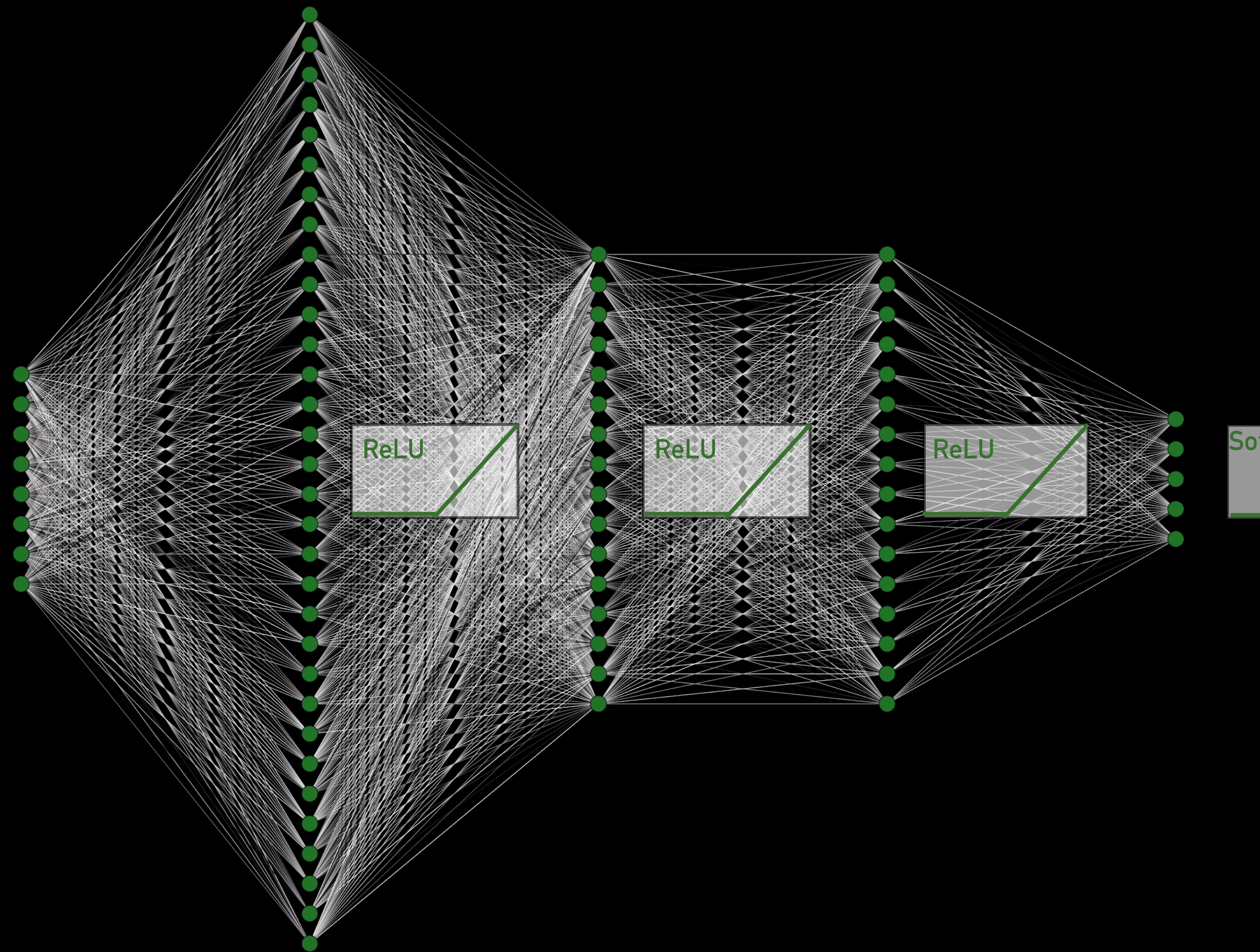
Quantization-aware training



Forward pass →



hls4ml + Google QKeras



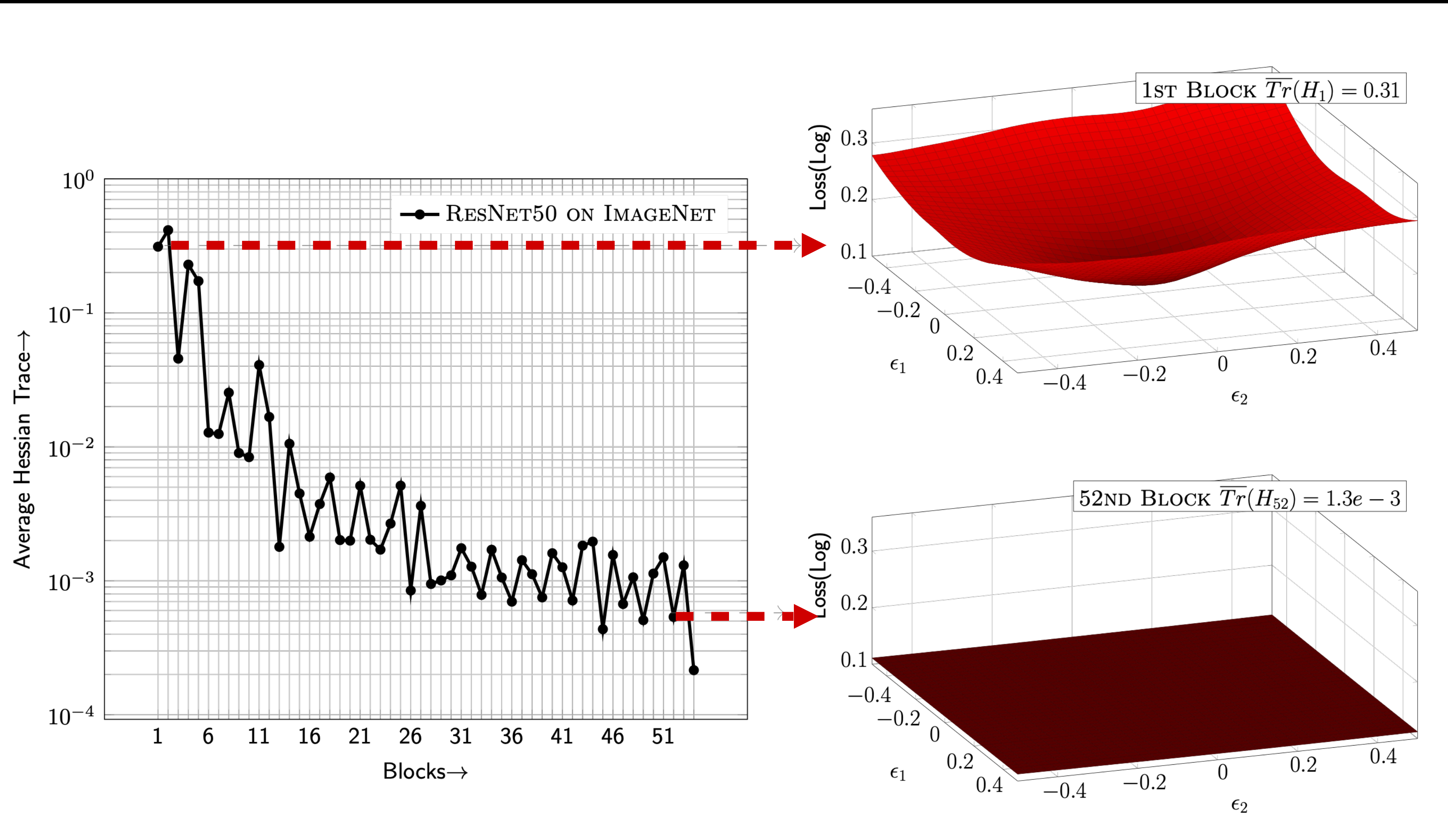
Forward pass →

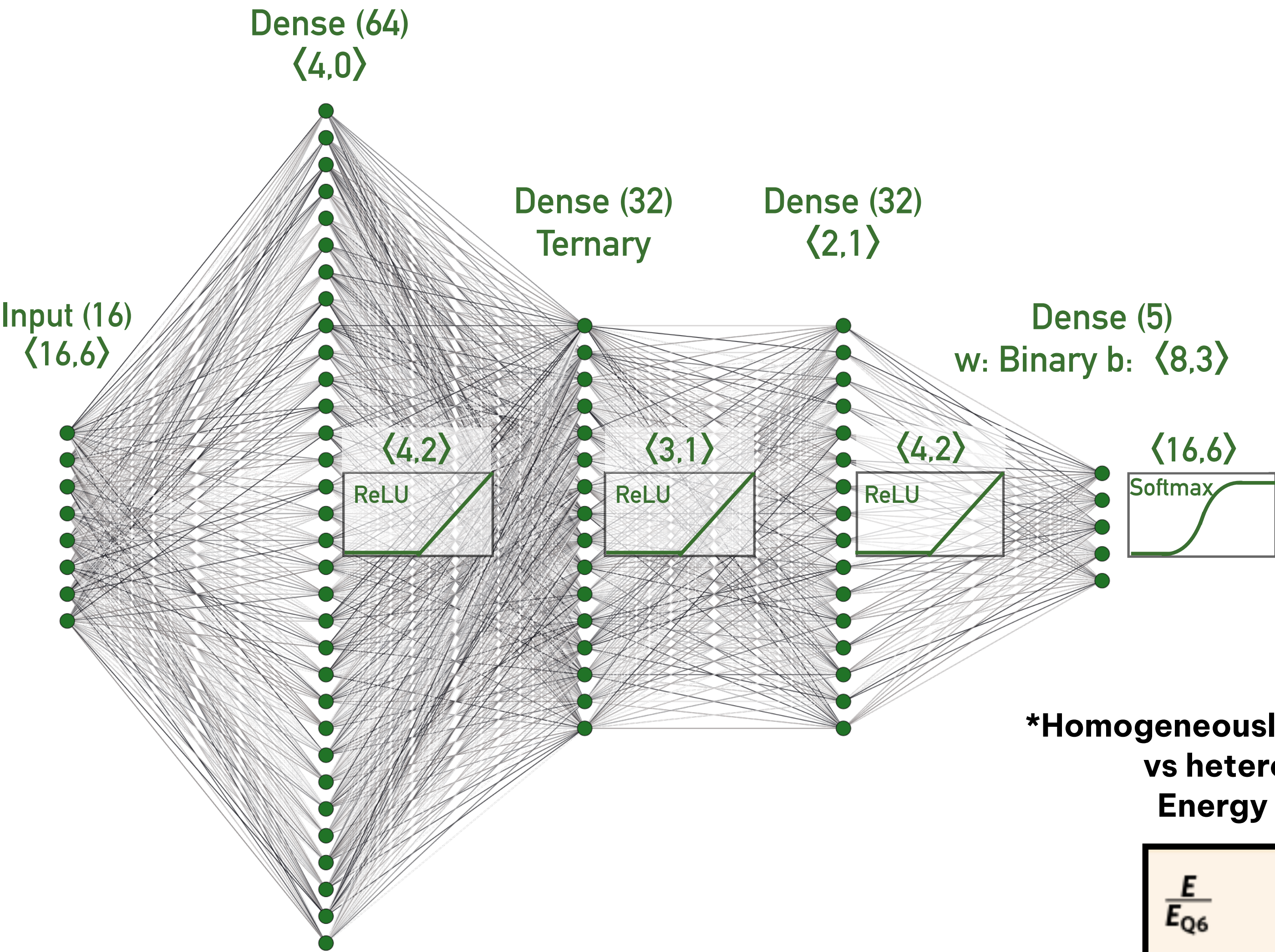
3.0

```
from tensorflow.keras.layers import Input, Activation
from qkeras import quantized_bits
from qkeras import QDense, QActivation
from qkeras import QBatchNormalization

x = Input((16))
x = QDense(64,
          kernel_quantizer = quantized_bits(6,0,alpha=1),
          bias_quantizer   = quantized_bits(6,0,alpha=1))(x)
x = QBatchNormalization()(x)
x = QActivation('quantized_relu(6,0)')(x)
x = QDense(32,
          kernel_quantizer = quantized_bits(6,0,alpha=1),
          bias_quantizer   = quantized_bits(6,0,alpha=1))(x)
x = QBatchNormalization()(x)
x = QActivation('quantized_relu(6,0)')(x)
x = QDense(32,
          kernel_quantizer = quantized_bits(6,0,alpha=1),
          bias_quantizer   = quantized_bits(6,0,alpha=1))(x)
x = QBatchNormalization()(x)
x = QActivation('quantized_relu(6,0)')(x)
x = QDense(5,
          kernel_quantizer = quantized_bits(6,0,alpha=1),
          bias_quantizer   = quantized_bits(6,0,alpha=1))(x)
x = Activation('softmax')(x)
```


Some layer more accommodating to aggressive quantization than others!





***Homogeneously quantized 6 bit model
vs heterogeneous model
Energy cost, chip area**

$\frac{E}{E_{Q6}}$	$\frac{\text{Bits}}{\text{Bits}_{Q6}}$
0.27	0.18*

But why stop there?

```
from tensorflow.keras.layers import Input
from HGQ import HQuantize, HDense

inp = Input((16,))
out = HQuantize(name='inp_q', beta=beta)(out)
out = HDense(64, activation='relu', beta=beta)(out)
out = HDense(32, activation='relu', beta=beta)(out)
out = HDense(32, activation='relu', beta=beta)(out)
out = HDense(5, activation='linear', beta=beta)(out)

hgq_model = Model(inp, out)
```

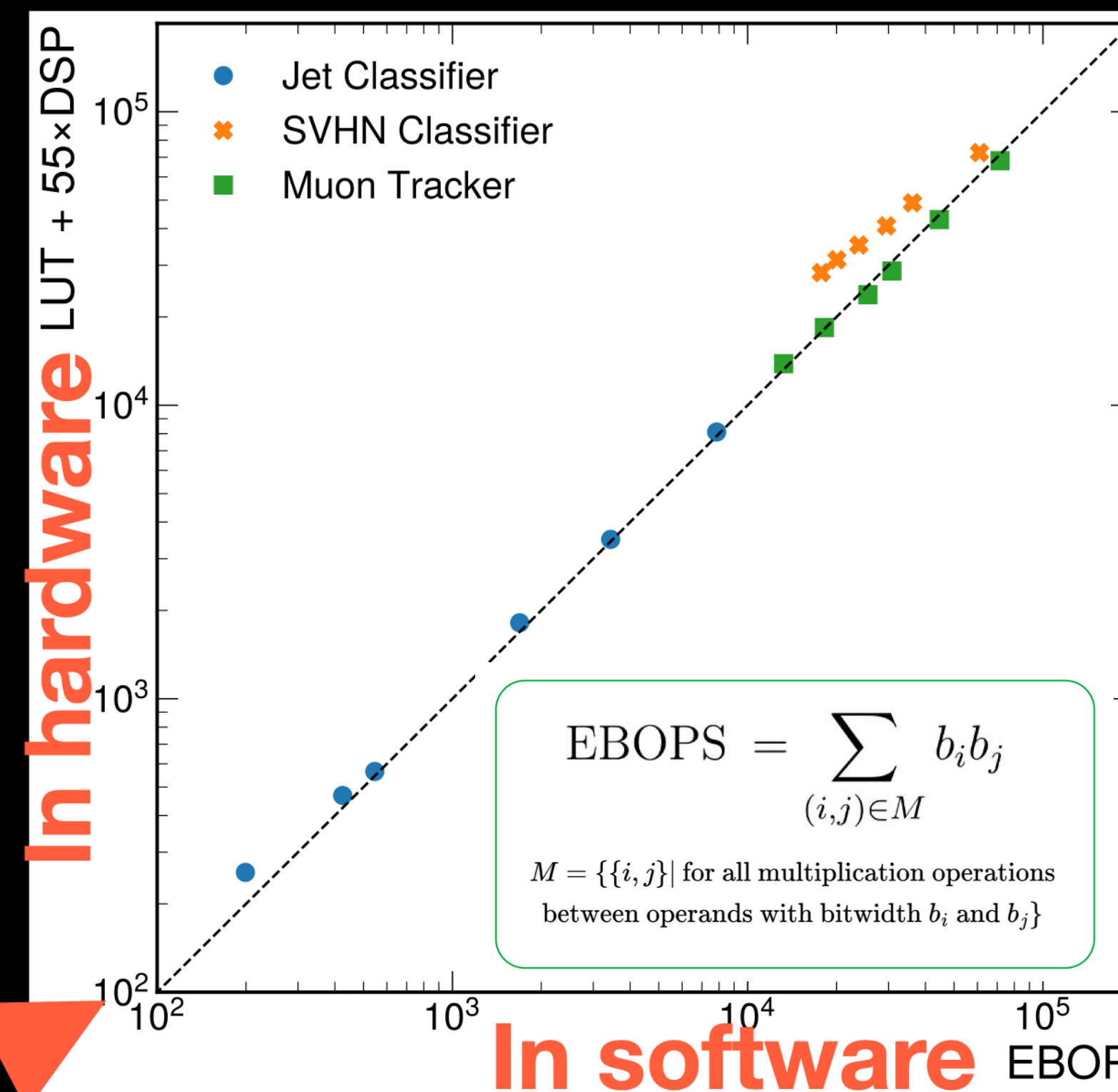

HGQ: High Granularity Quantization

C.Sun et al. 2024

```
from tensorflow.keras.layers import Input
from HGQ import HQuantize, HDense

inp = Input((16,))
out = HQuantize(name='inp_q', beta=beta)(out)
out = HDense(64, activation='relu', beta=beta)(out)
out = HDense(32, activation='relu', beta=beta)(out)
out = HDense(32, activation='relu', beta=beta)(out)
out = HDense(5, activation='linear', beta=beta)(out)

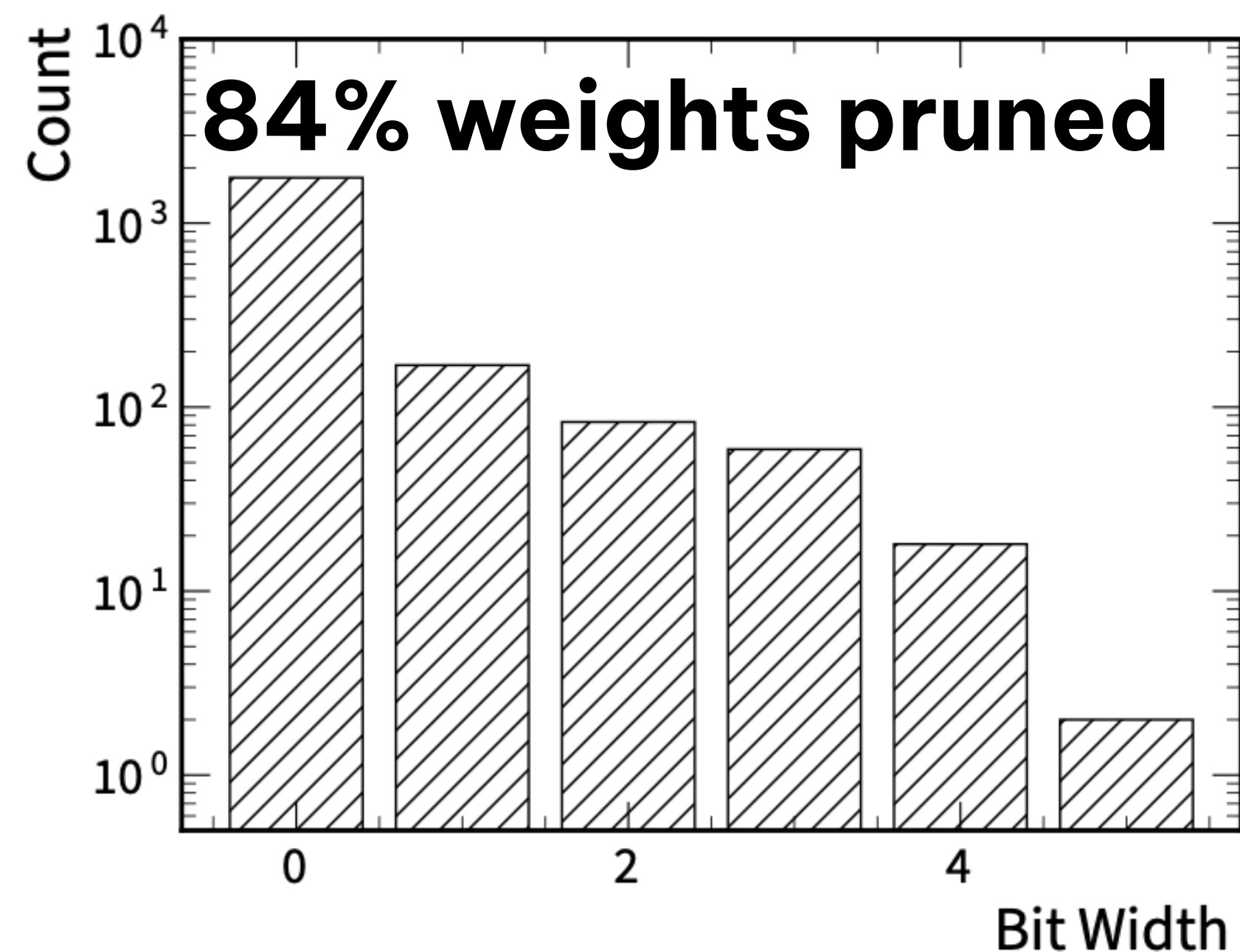
hgq_model = Model(inp, out)
```



Proxy for resource cost

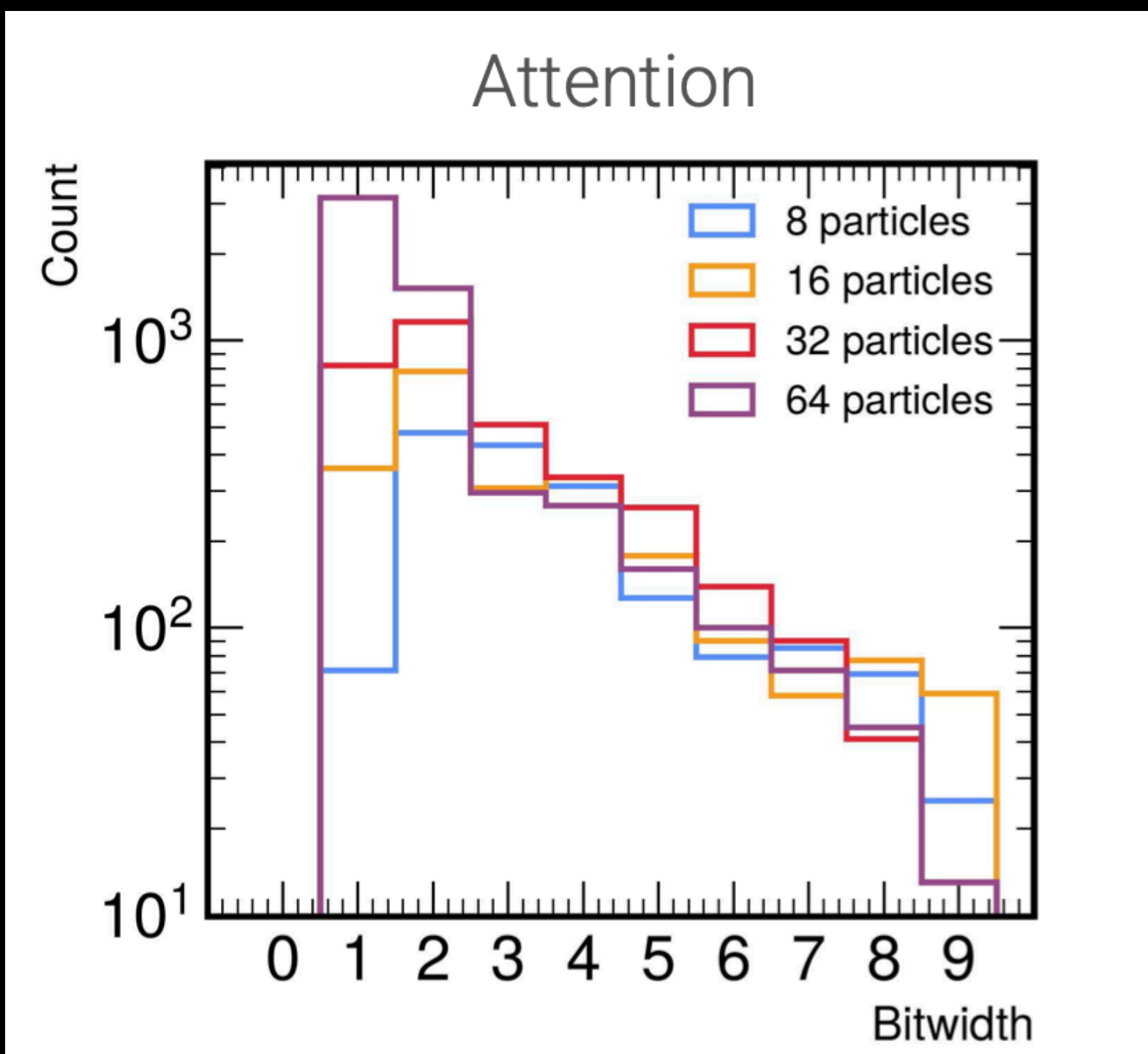
$$\mathcal{L} = \mathcal{L}_{\text{base}} + \beta \cdot \overline{\text{EBOPS}} + \gamma \cdot \text{L1}_{\text{norm}}$$

Making bitwidths differentiable! Fully heterogeneous quantization!



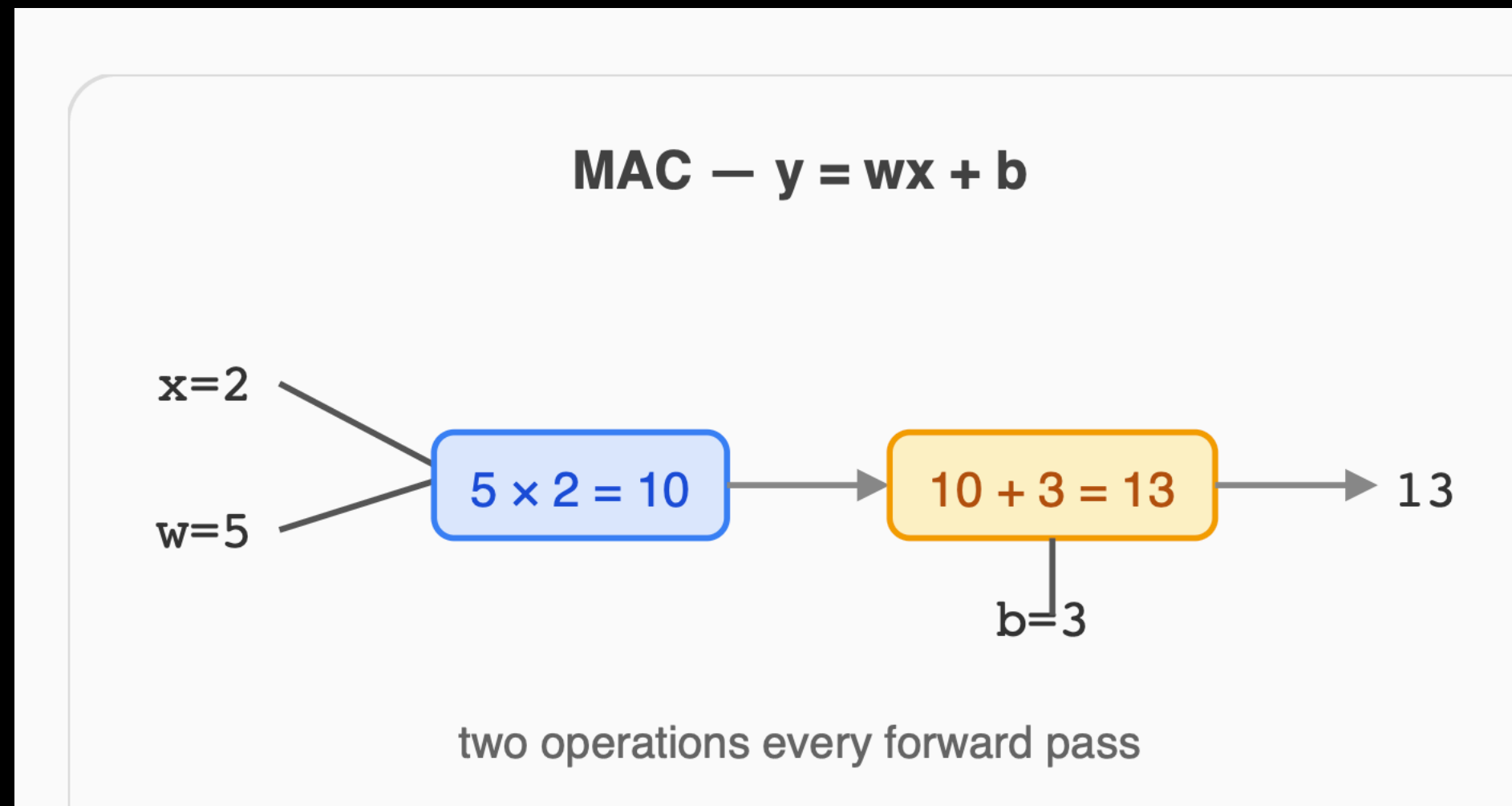
**It's not a FAD:
first results in using Flows for unsupervised
Anomaly Detection at 40 MHz at the Large Hadron
Collider**

**Normalizing Flows on
chip with 35 ns latency**

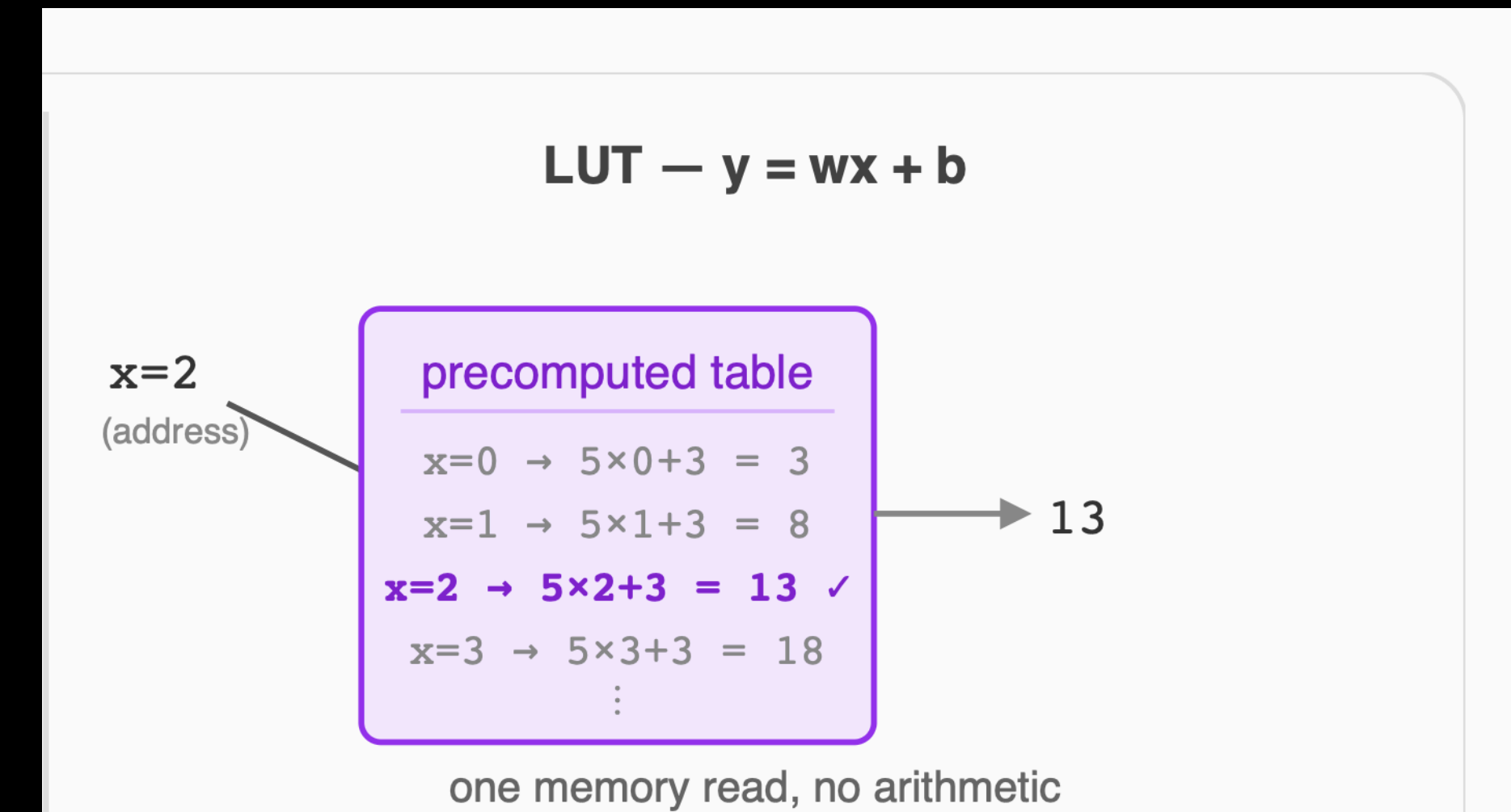


**Heterogeneously
quantized
transformers with 100
ns latency!**

Multiply-Accumulate



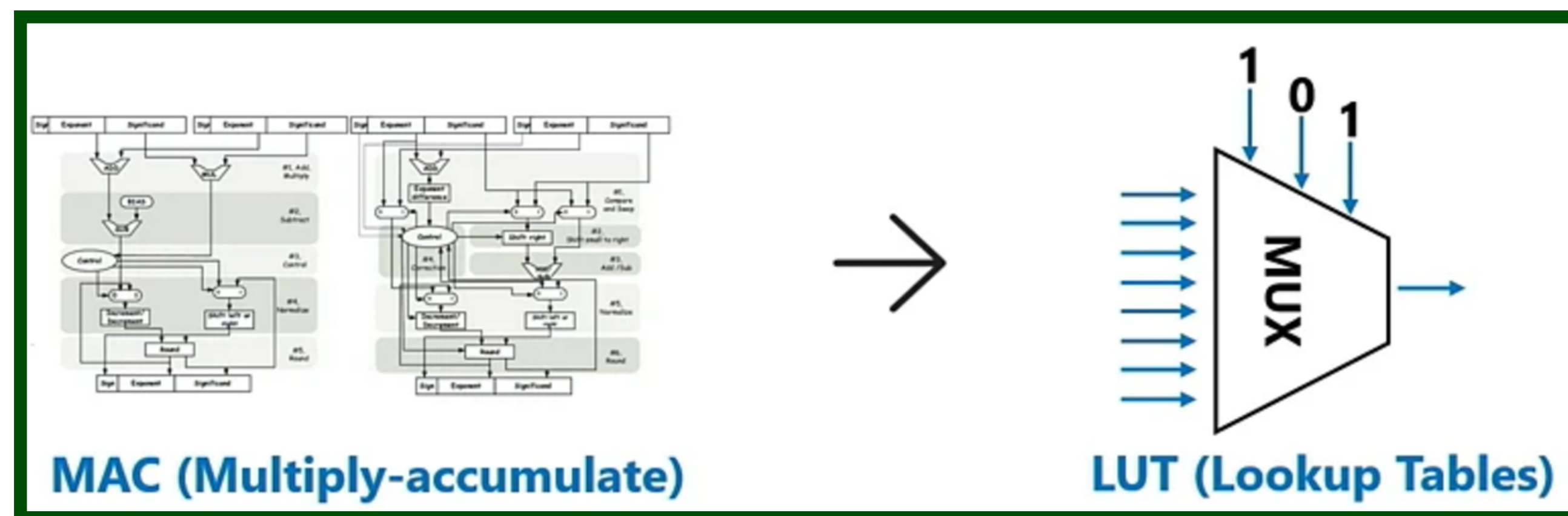
Lookup Table



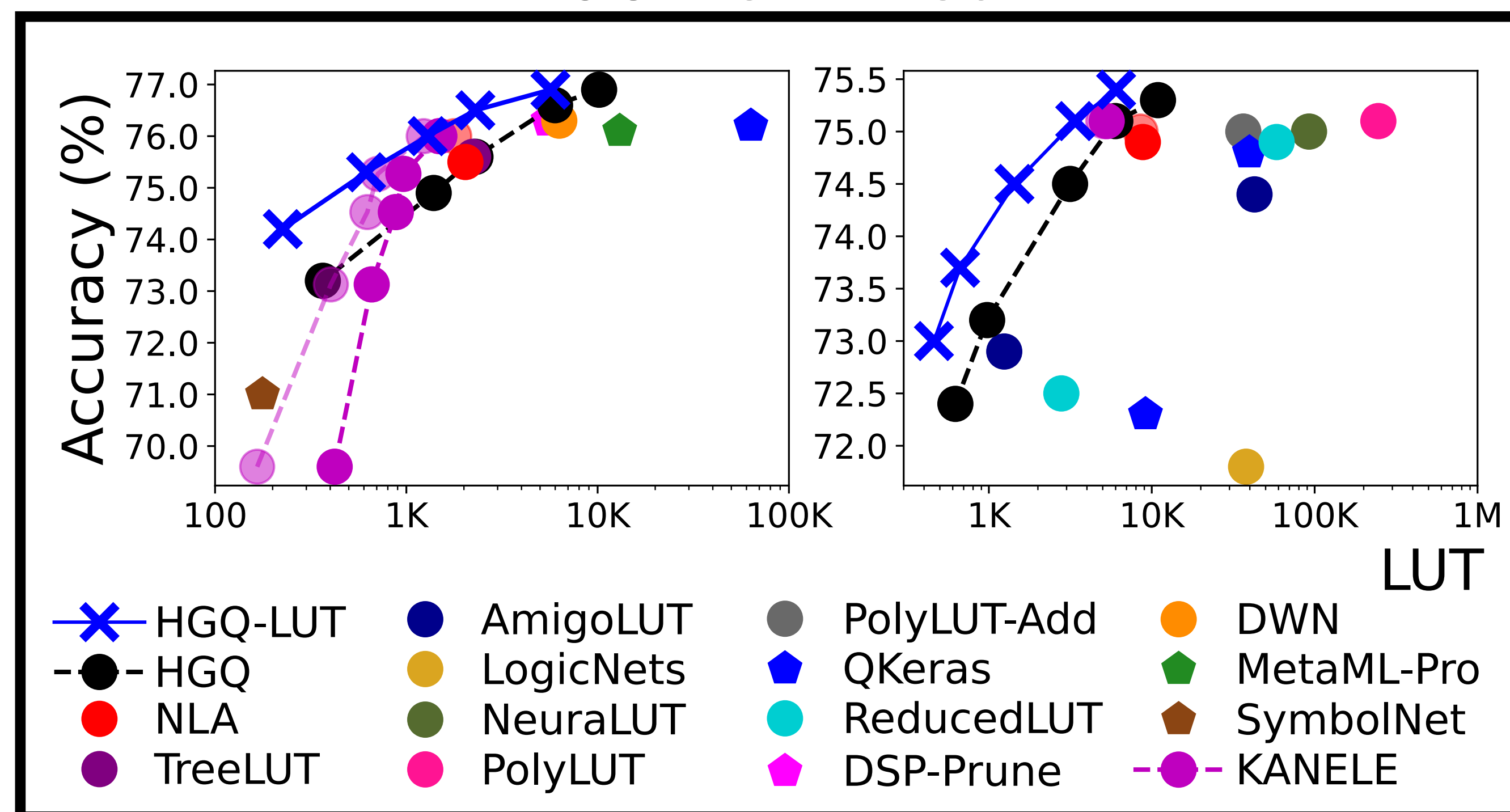
Store precomputed results for retrieval rather than expensive runtime compute
Enter: LUTnets

C. Sun et al (2026)

C. Sun et al (2026)



Blue is LUT dense layers
Black is without!



Why do tree-based models still outperform deep learning on typical tabular data?

Leo Grinsztajn, Edouard Oyallon, Gael Varoquaux

06 Jun 2022 (modified: 16 Jan 2023) NeurIPS 2022 Datasets and Benchmarks Readers:  Everyone [Show Bibtex](#) [Show Revisions](#)

Abstract: While deep learning has enabled tremendous progress on text and image datasets, its superiority on tabular data is not clear. We contribute extensive benchmarks of standard and novel deep learning methods as well as tree-based models such as XGBoost and Random Forests, across a large number of datasets and hyperparameter combinations. We define a standard set of 45 datasets from varied domains with clear characteristics of tabular data and a benchmarking methodology accounting for both fitting models and finding good hyperparameters. Results show that tree-based models remain state-of-the-art on medium-sized data ($\sim 10K$ samples) even without accounting



Computer Science > Machine Learning

[Submitted on 11 Oct 2022 (v1), last revised 25 Oct 2022 (this version, v3)]

Neural Networks are Decision Trees

Caglar Aytekin

In this manuscript, we show that any neural network with any activation function can be represented as a decision tree. The representation is equivalence and not an approximation, thus keeping the accuracy of the neural network exactly as is. We believe that this work provides better understanding of neural networks and paves the way to tackle their black-box nature. We share equivalent trees of some neural networks and show that besides providing interpretability, tree representation can also achieve some computational advantages for small networks. The analysis holds both for fully connected and convolutional networks, which may or may not also include skip connections and/or normalizations.

Subjects: **Machine Learning (cs.LG)**

Cite as: [arXiv:2210.05189](#) [cs.LG]

(or [arXiv:2210.05189v3](#) [cs.LG] for this version)

<https://doi.org/10.48550/arXiv.2210.05189> 

Submission history

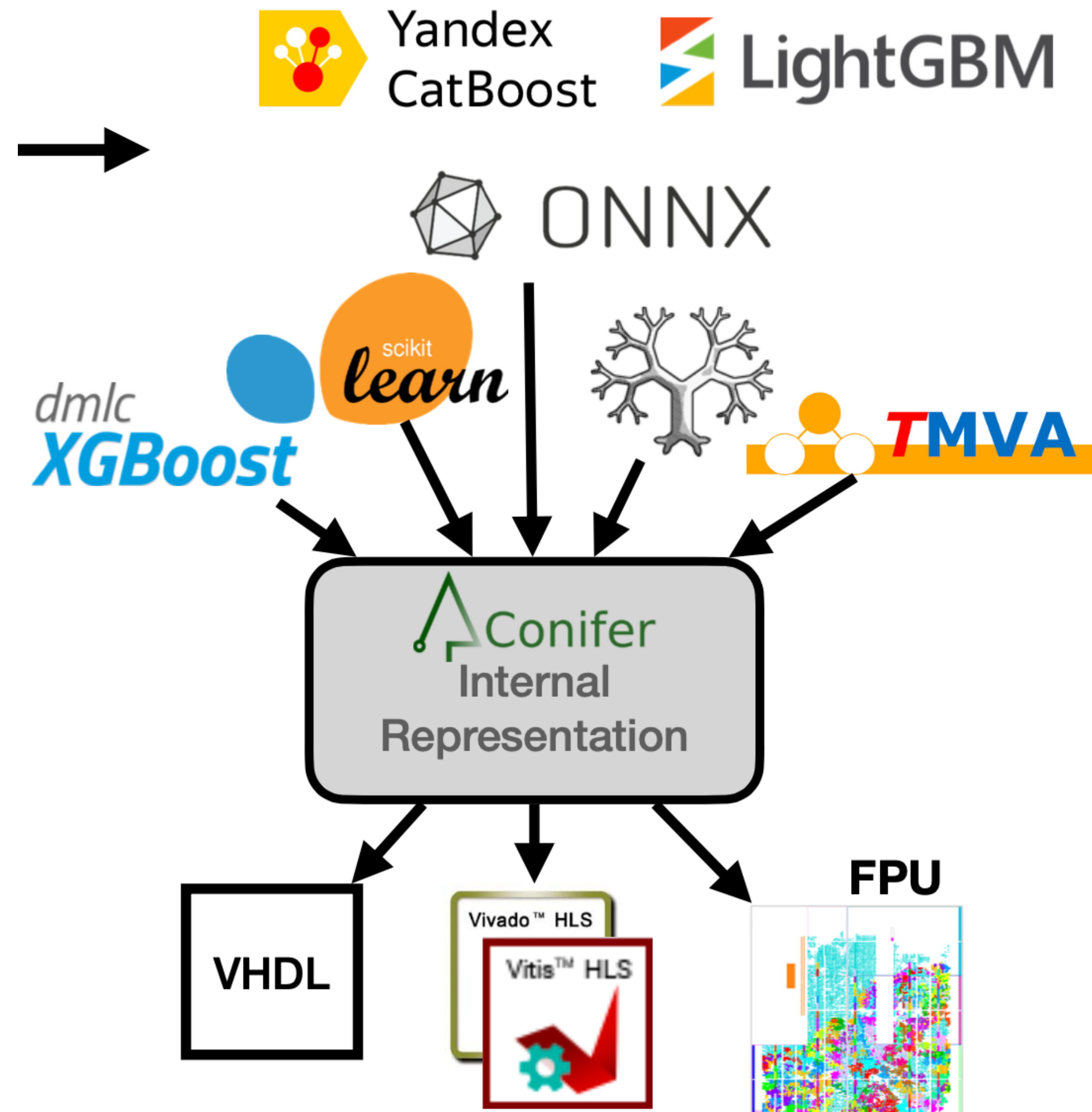
From: Çağlar Aytekin [[view email](#)]

[v1] Tue, 11 Oct 2022 06:49:51 UTC (216 KB)

[v2] Mon, 17 Oct 2022 15:18:14 UTC (224 KB)

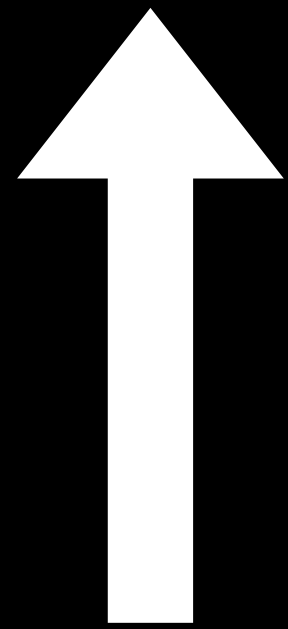
[v3] Tue, 25 Oct 2022 17:32:33 UTC (240 KB)

%VU9P	Accuracy	Latency	DSP	LUT
qDNN	75.6%	40 ns	22 (~0%)	1%
BDT	74.9%	5 ns	-	0.5%

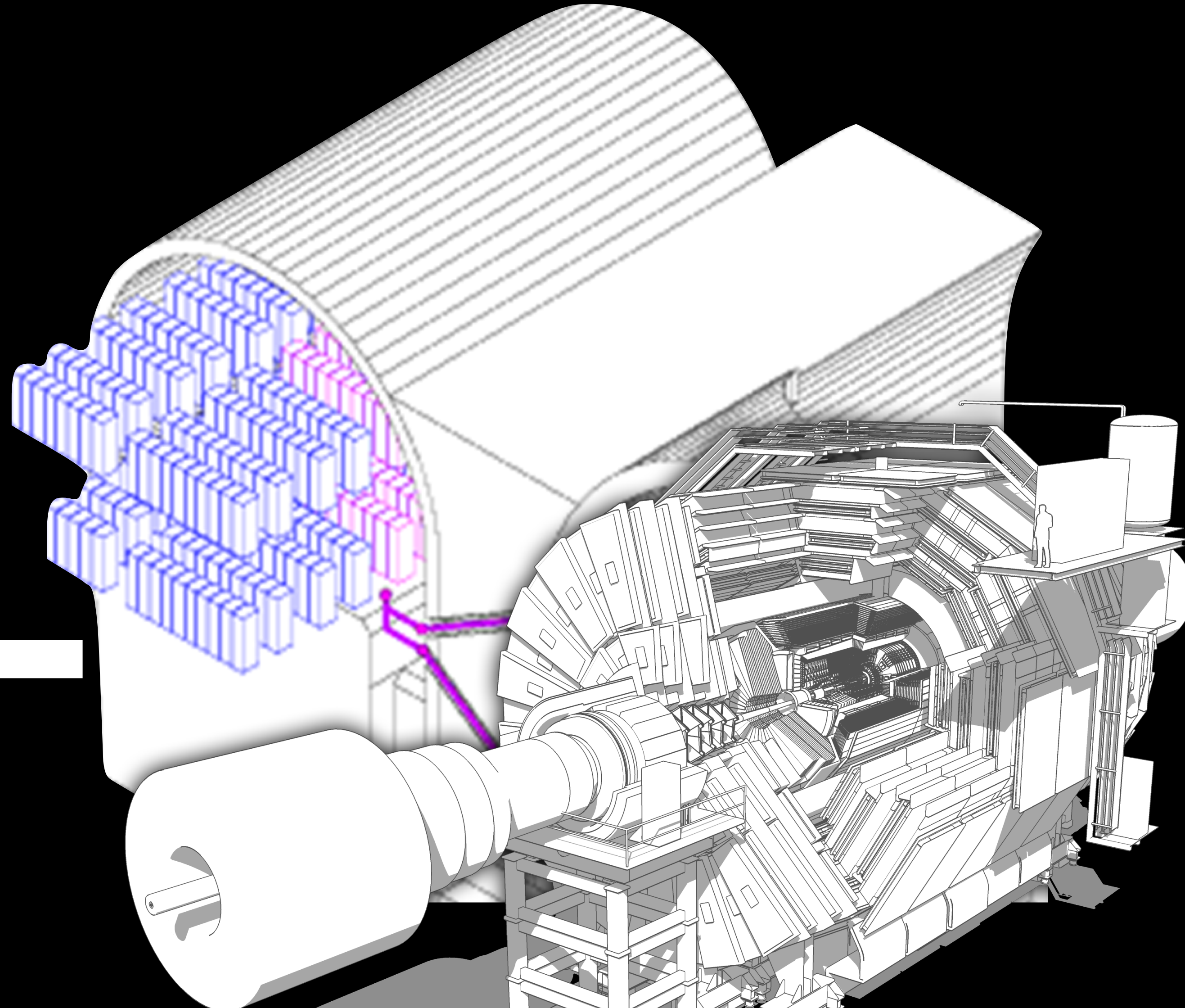


40 MHz datascouting: Acquisition and real time analysis of all trigger data on AIEs and GPU

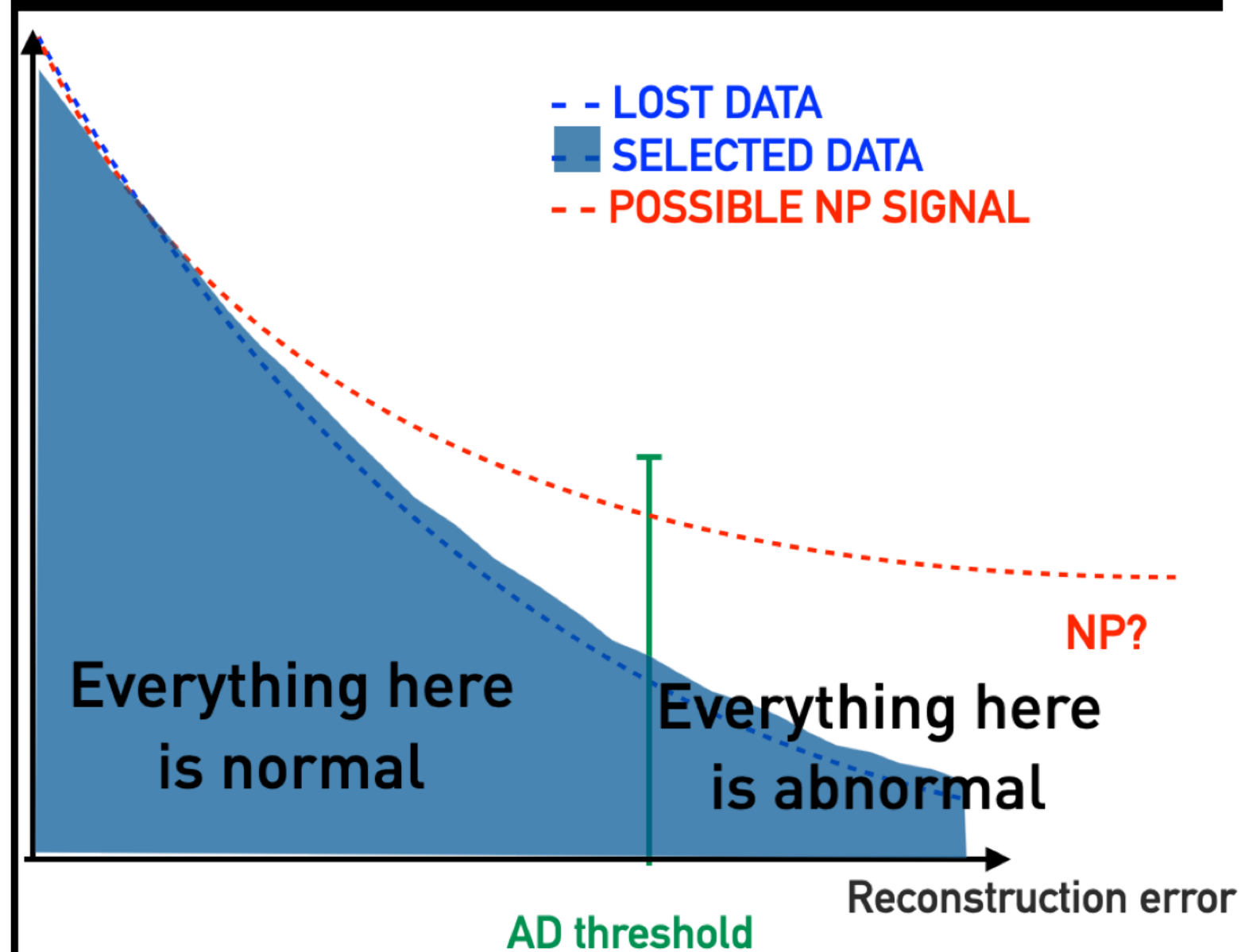
On-GPU/ AI engine real-time analysis in $O(1) \mu s$,
to reduce rate to 5 GB/s!



On-FPGA processing on
7.65 Tb/second of reconstructed
trigger objects

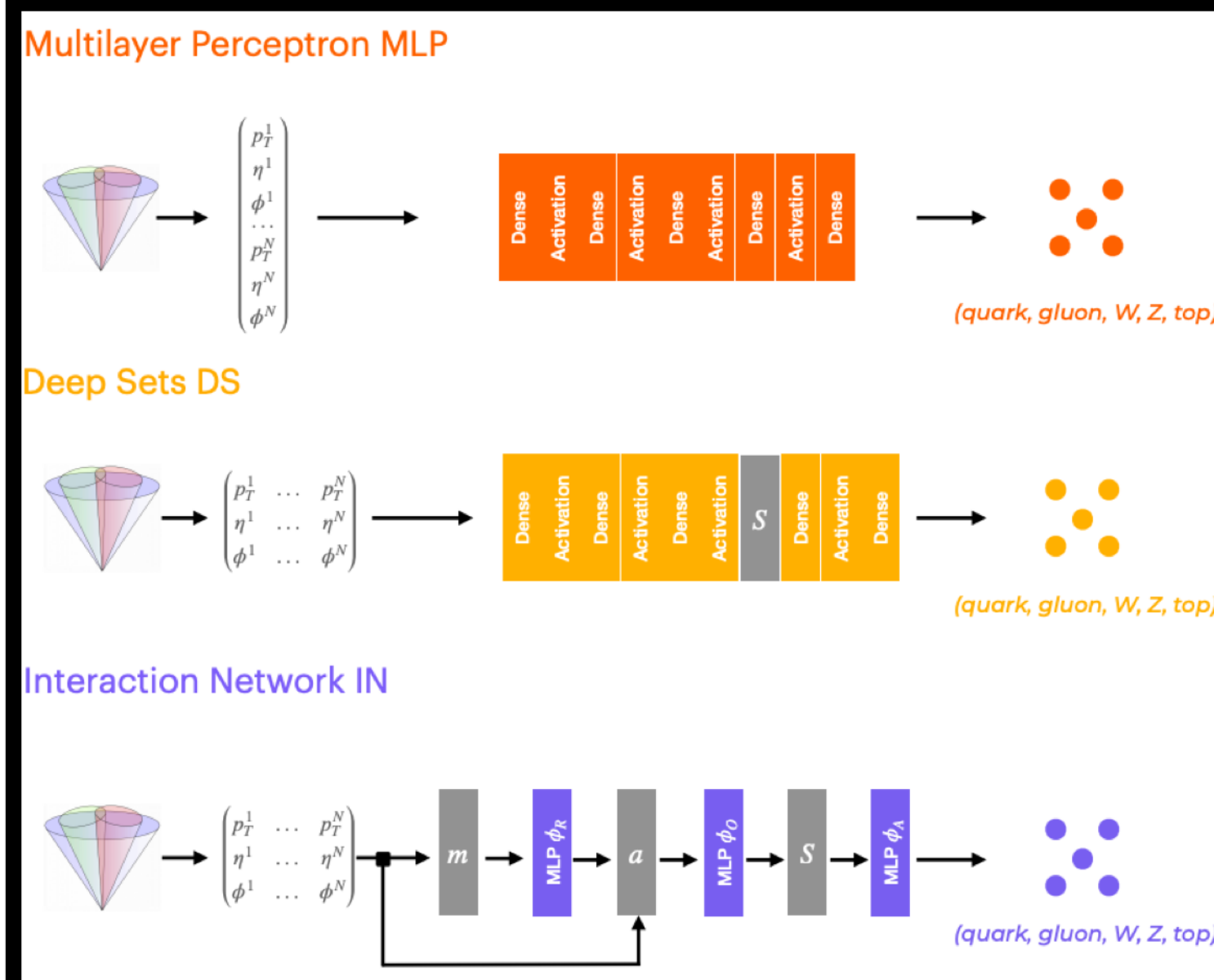


Anomaly detection with VAEs in 50 ns



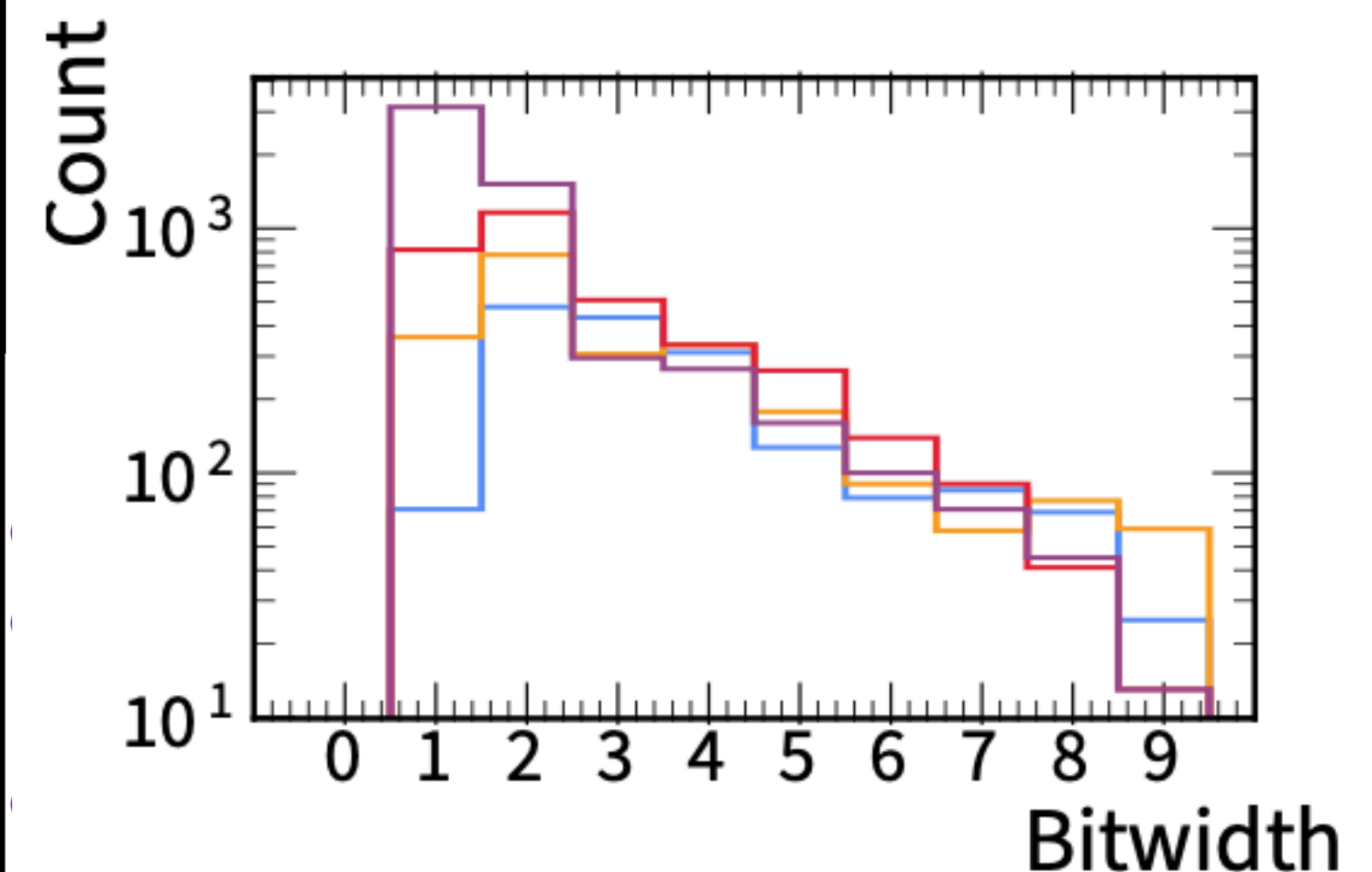
CMS DP2023_079
E. Govorkova et al (2022)

Quantised Interaction Networks and Deep Sets in <160 ns



P. Odagiu et al. 2024

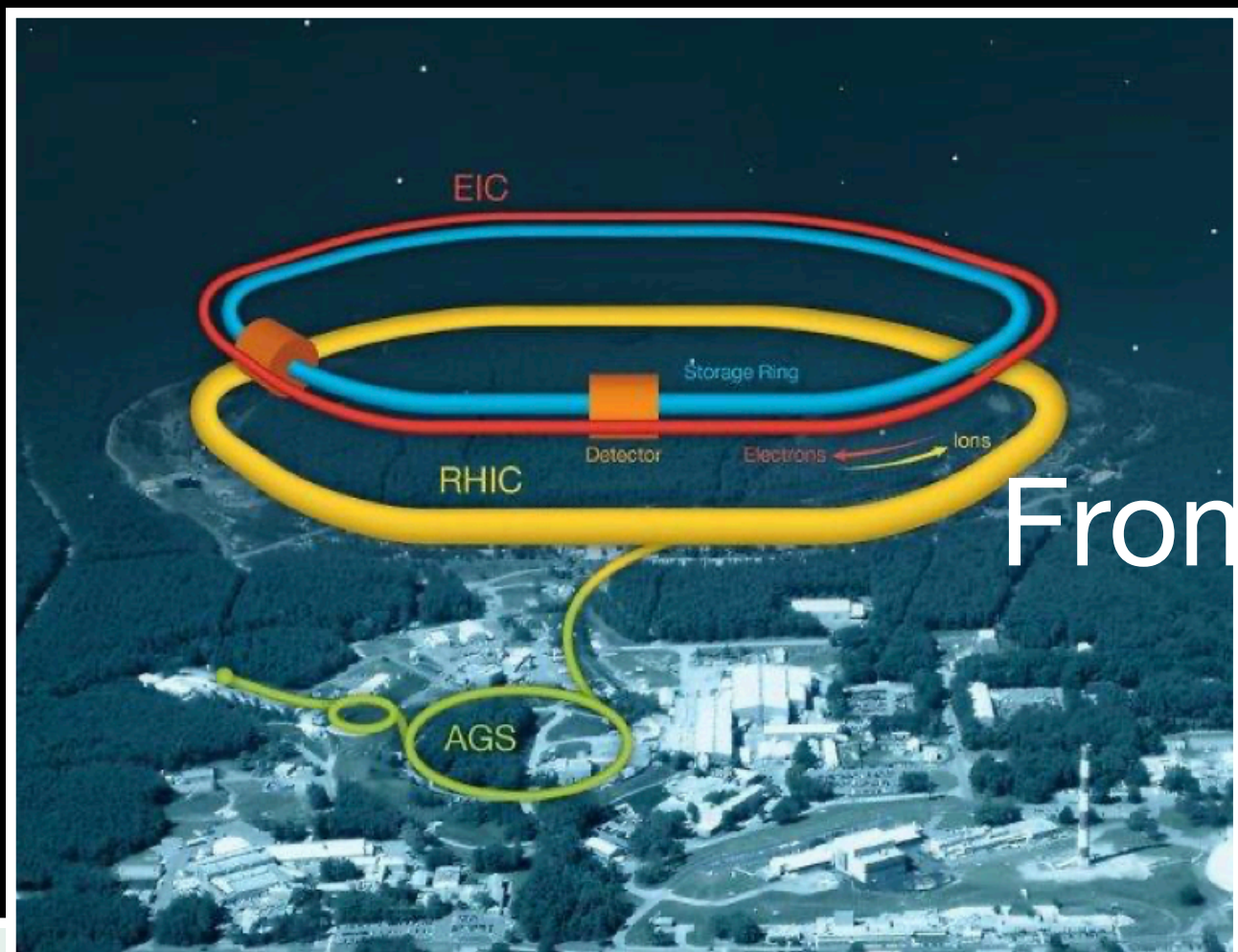
Fully heterogeneously quantized transformers in <100 ns



(a) Attention Layers

[arxiv:2510.24784](https://arxiv.org/abs/2510.24784)

Multi-FPGA NNs for real time data reduction ePIC at the EIC



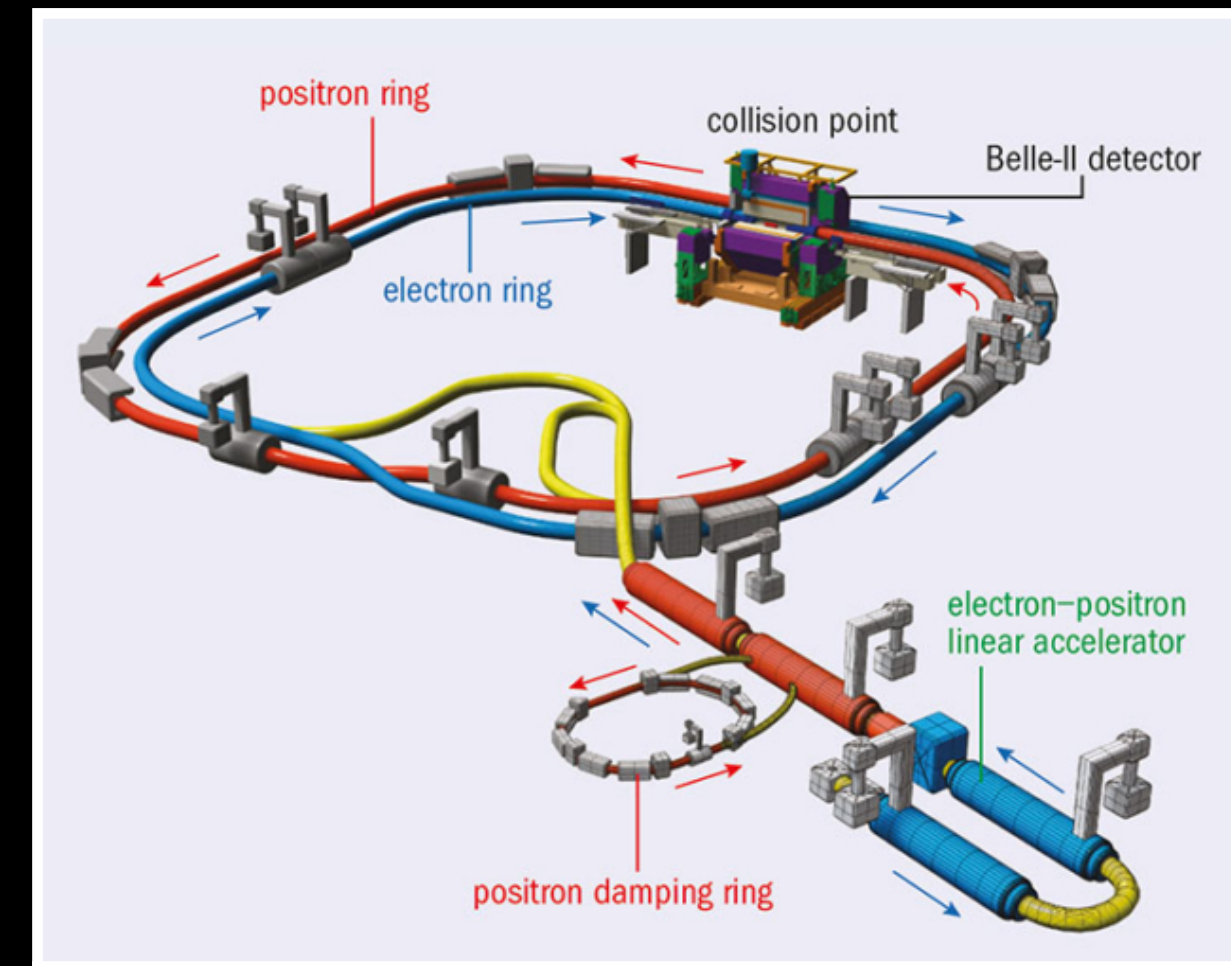
From New York...

- The NN model mimics the DAQ system architecture
 - 30 (# of subsectors x # of sectors) **DAM MLP networks** deployed on 30 **DAM FPGAs**.
 - 6 sector **MLP networks + 1 aggregation network** deployed on the **TP FPGA**.

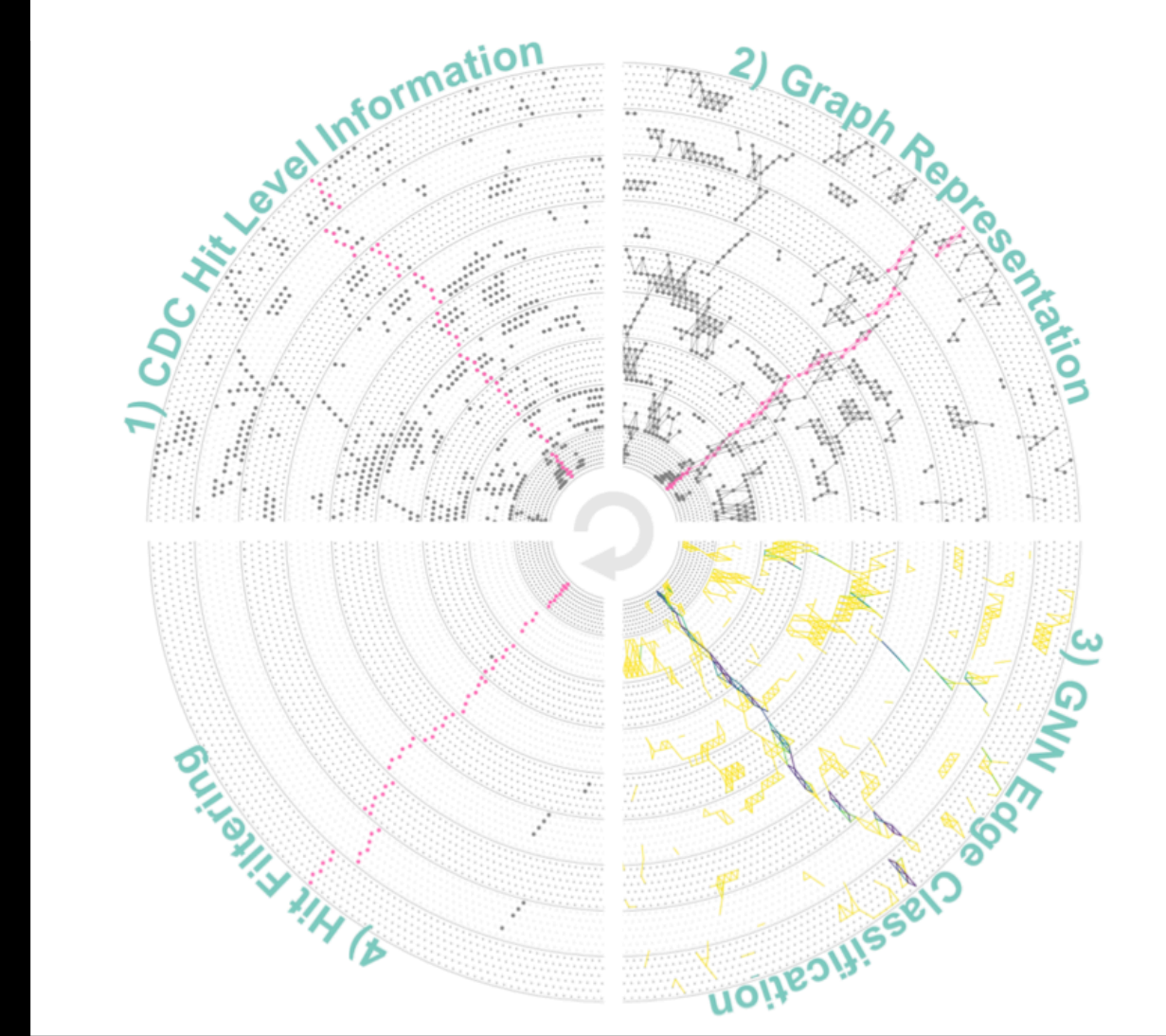
HGQ2+hls4ml

Rossi et. al.

Real-time GNNs on FPGAs for clustering Belle II at SuperKEKB



to Japan!



QKeras+custom

Haide et. al.

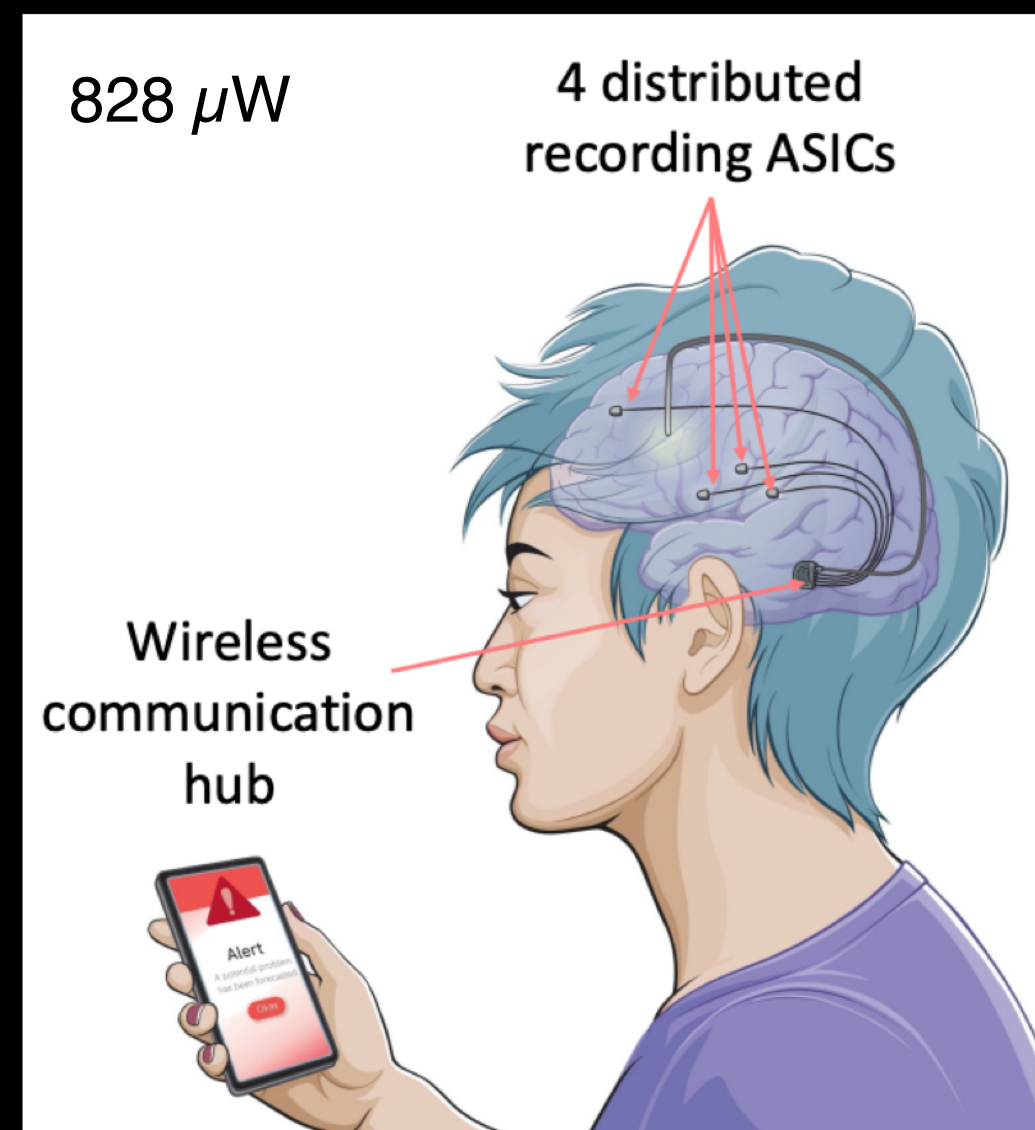
...and outside

Semantic segmentation for autonomous vehicles



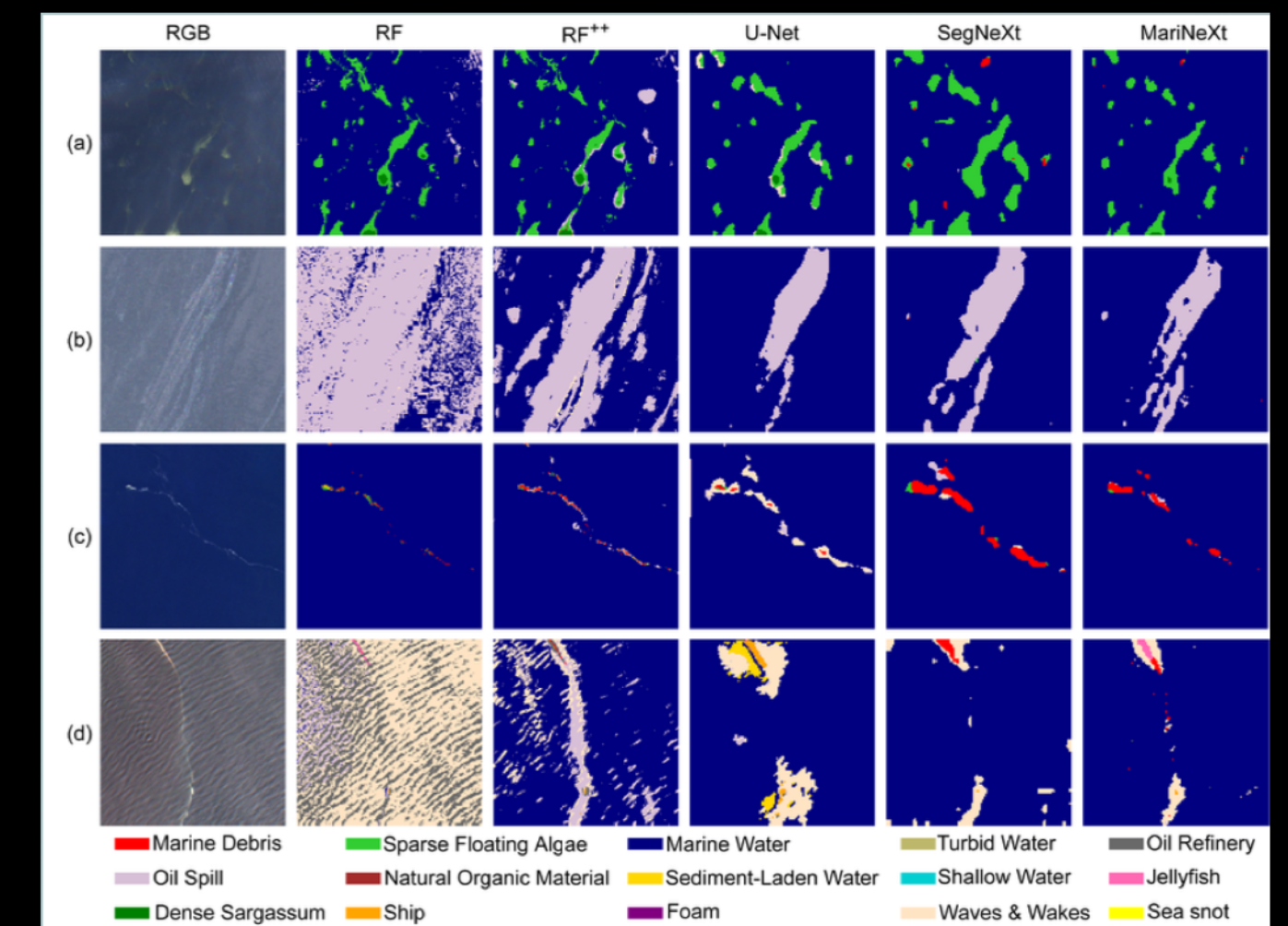
N. Ghielmetti et al. 2022

Seizure Predicting Brain Implant



W. Lemaire et al. 2022

Earth monitoring in satellites

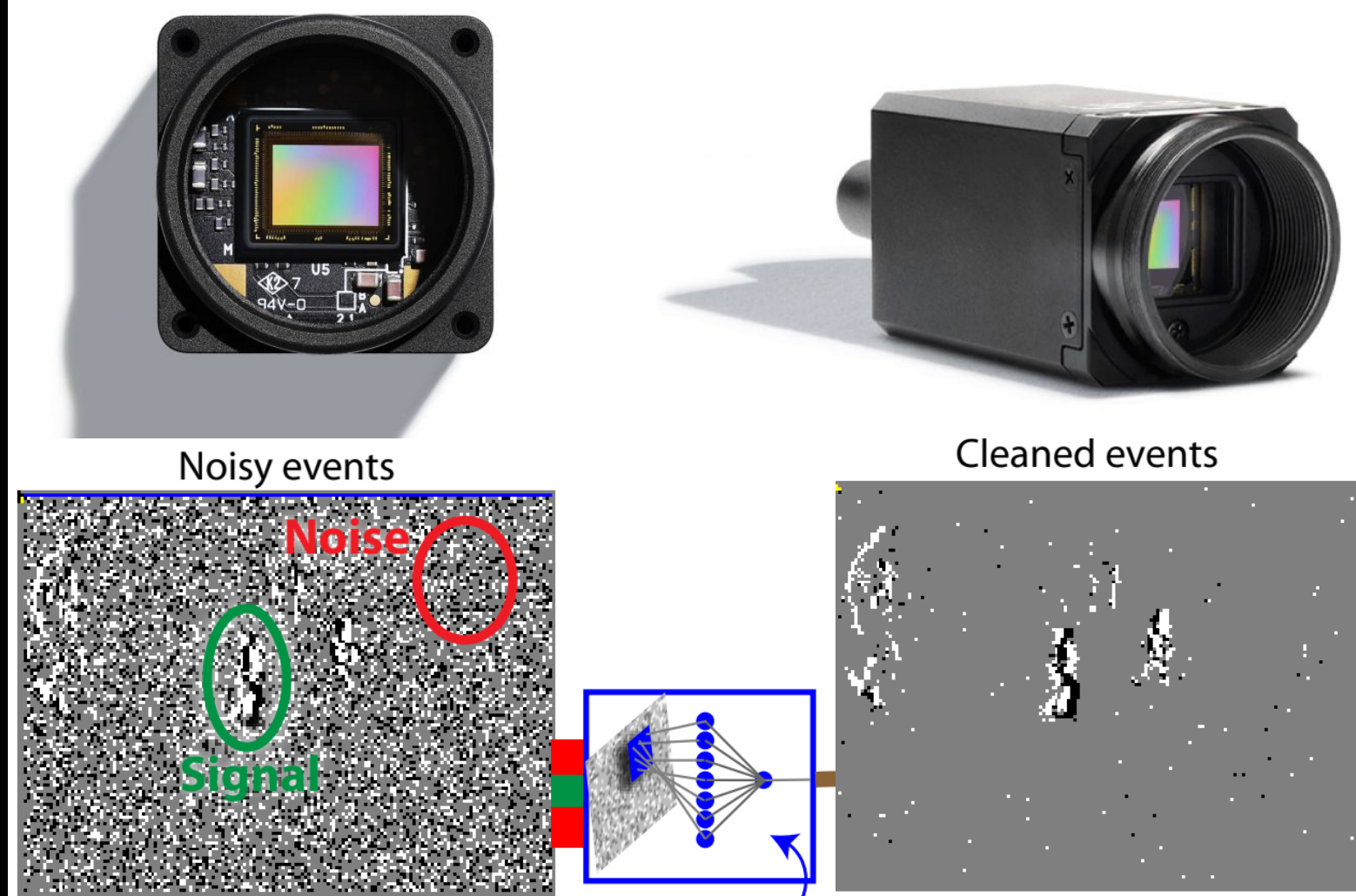


Edge SpAlce, S. Summers

- MLPerf tinyML benchmarking
- For fusion science phase/mode monitoring
- Crystal structure detection
- Triggering in DUNE

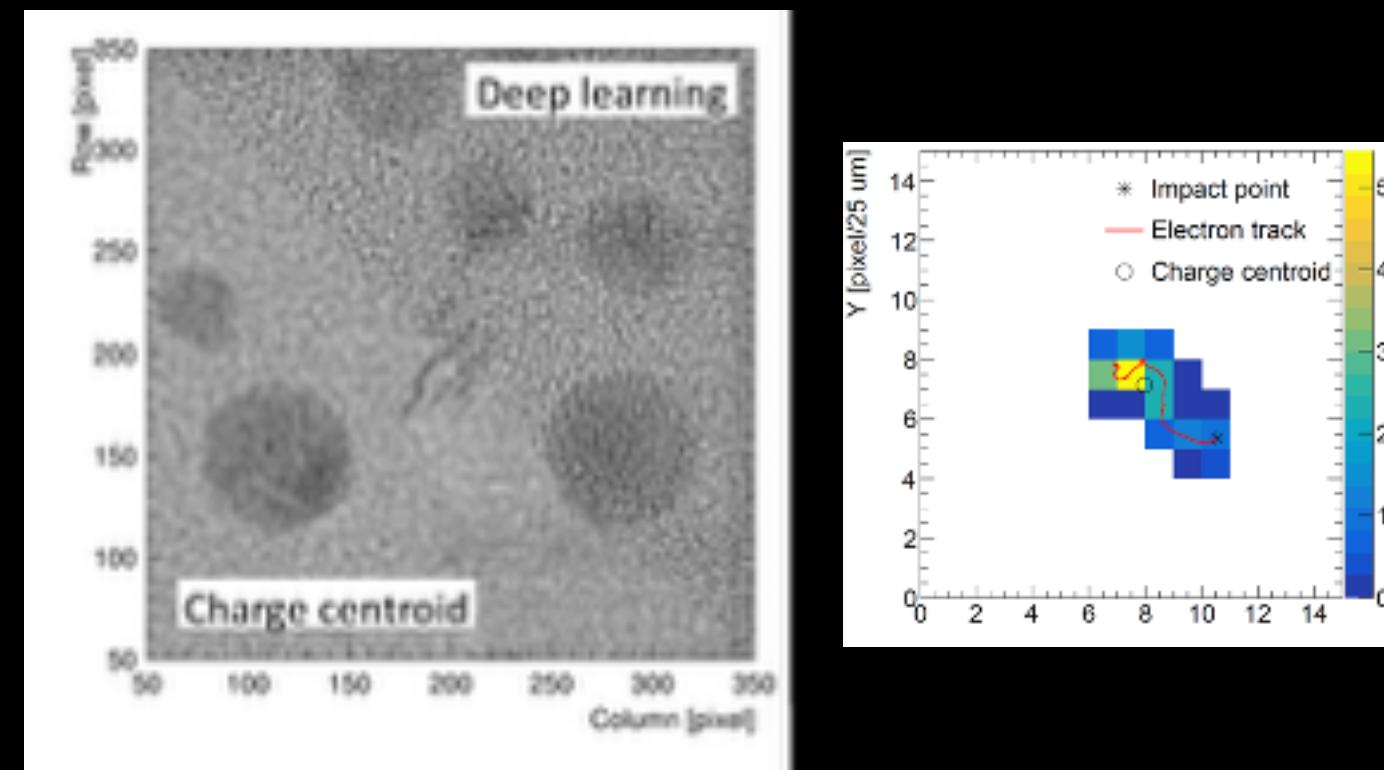
- Accelerator control
- Magnet Quench Detection
- Food contamination detection
- Quantum control etc....

Denoising event cameras



2023 CVPRW Workshop on Event-Based Vision

Electron microscopy sub-pixel resolution



L. Asfour, B. Ramhorst, Ahash
Thamilmaran (ETHZ), PSI, SDSC

Speech enhancement on edge devices

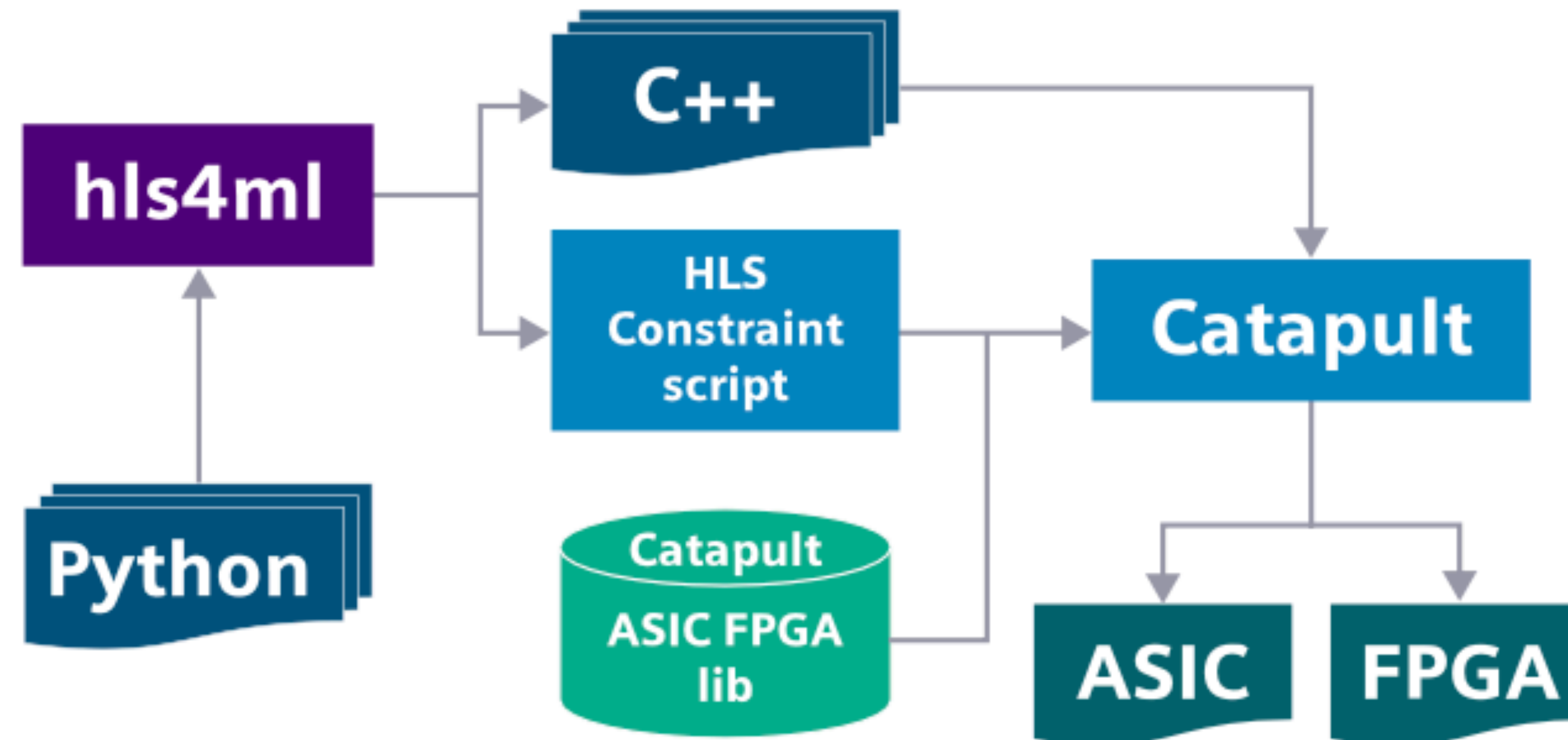


ETHZ, GN (Jabra)

PRESS RELEASE

Siemens simplifies development of AI accelerators for advanced system-on-chip designs with Catapult AI NN

May 21, 2024
Plano, Texas



“...brings together hls4ml and Siemens' Catapult™ HLS software...
... addresses the unique requirements of
ML accelerator design for power, performance, and area on custom silicon”

fast machine learning foundation

fastmachinelearning.org

Collaborating Institutes

Argonne National Laboratory

CERN

California Institute of Technology

Carnegie Mellon University

ETH Zurich

Fermi National Accelerator Laboratory

Georgia Institute of Technology

HEPIA - Haute école du paysage, d'ingénierie et d'architecture

Imperial College London

Jefferson Lab

Jožef Stefan Institute

KIT

LPNHE, CNRS/IN2P3, France

Los Alamos National Laboratory

Massachusetts Institute of Technology

National Yang Ming Chiao Tung University

Northwestern University

Norwegian University of Science and Technology

Pacific Northwest National Laboratory

Princeton University

Purdue University

SLAC

Southern Methodist University

Technical University of Denmark

Tohoku University

UC San Diego

UCL

University of Bristol

University of Chicago

University of Colorado Boulder

University of Geneva

University of Illinois at Urbana-Champaign

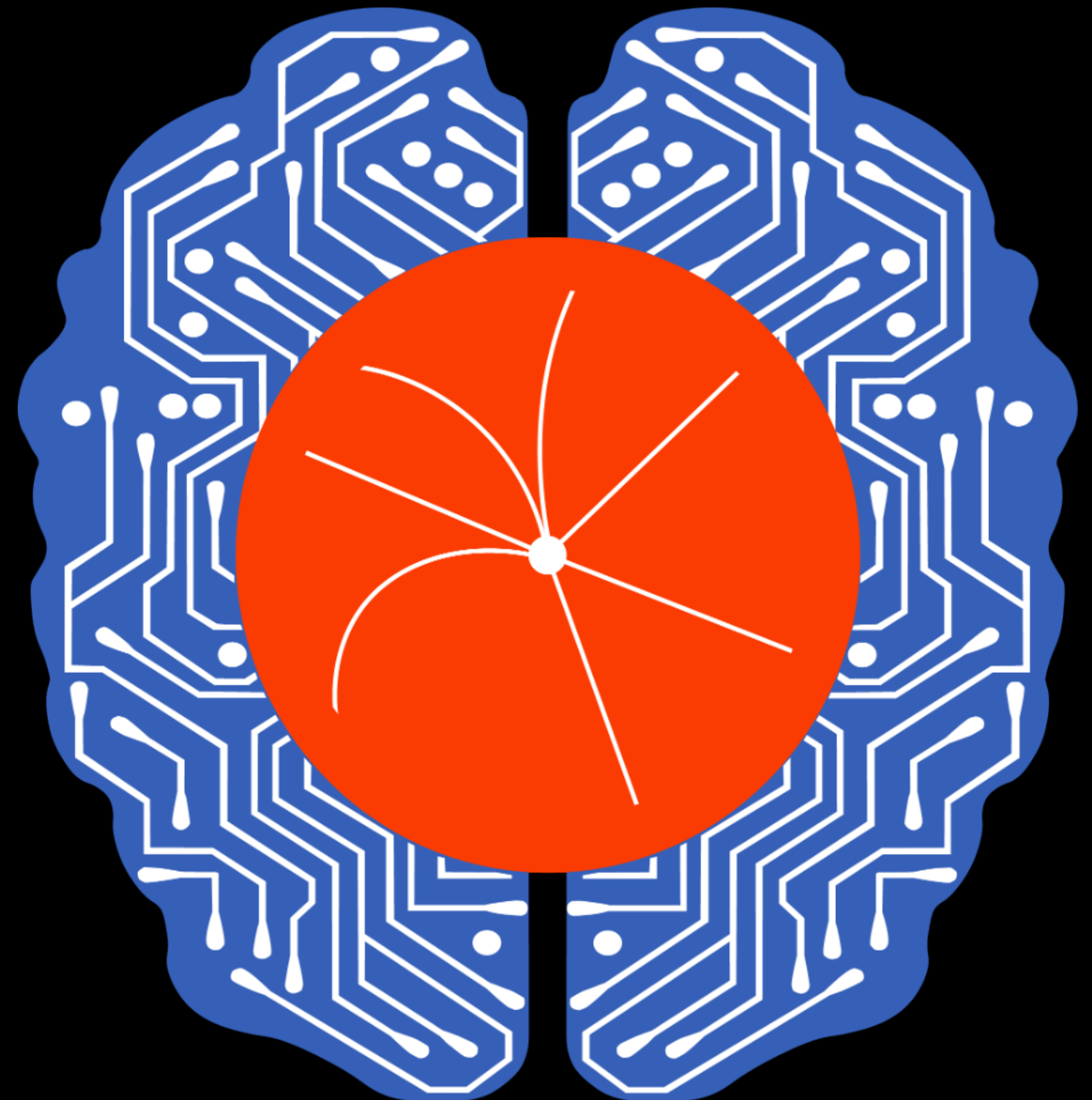
University of Michigan

University of Pennsylvania

University of Washington

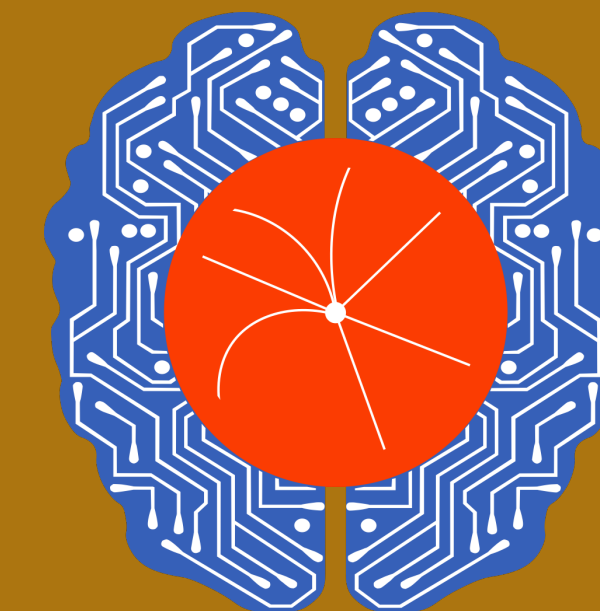
University of Wisconsin-Madison

Université de Sherbrooke



The Fast ML community

<https://fastmachinelearning.org/>



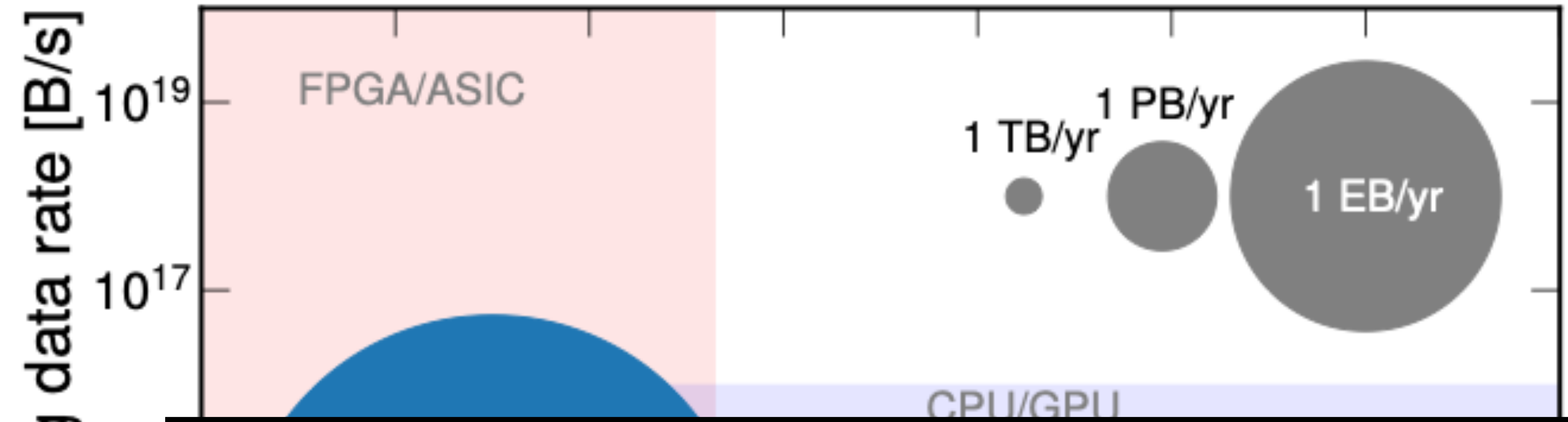
Join our Slack to get updates!

Backup

What looms on the horizon?



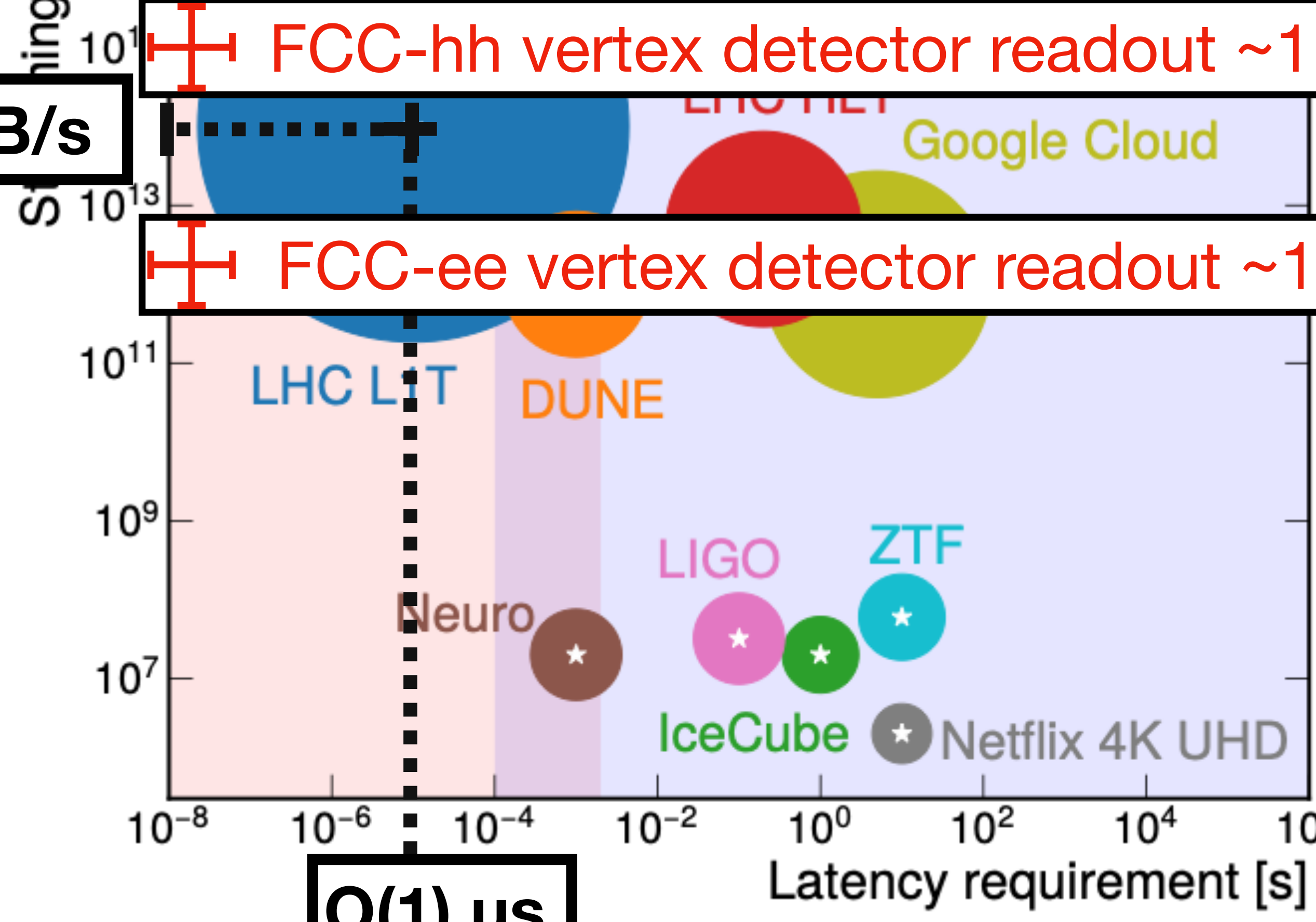
A3D3 Institute



FCC-hh vertex detector readout ~ 1 PB/s

O(100) TB/s

FCC-ee vertex detector readout ~ 1 TB/s



O(1) μs

Time to process



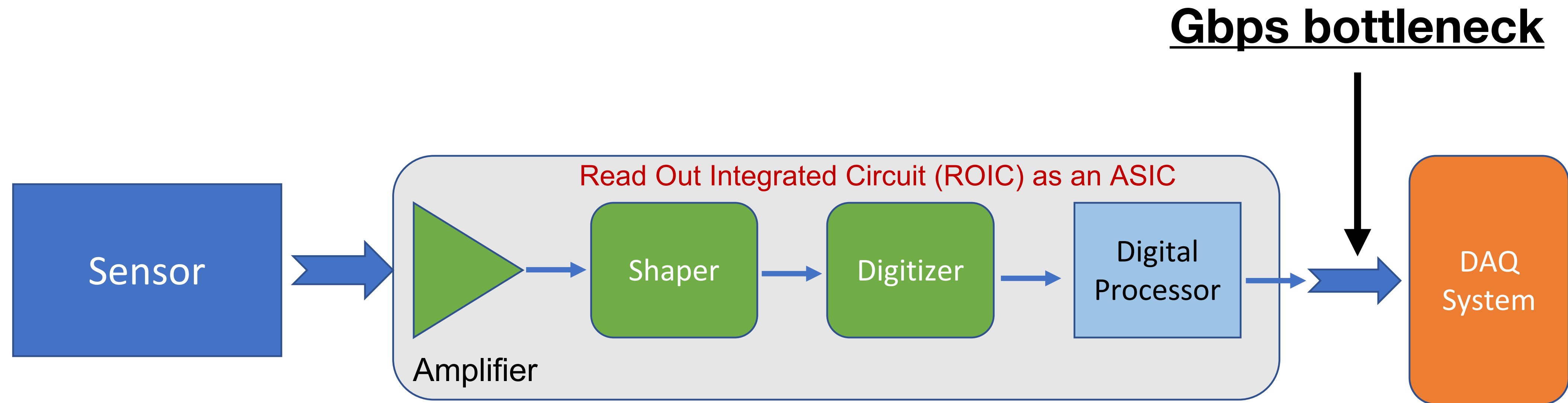
How much data goes through system



**This readout technology doesn't (yet)
exist!**

**Ultimate precision at future colliders
could be bottlenecked by our data
acquisition systems**

Data readout bottleneck



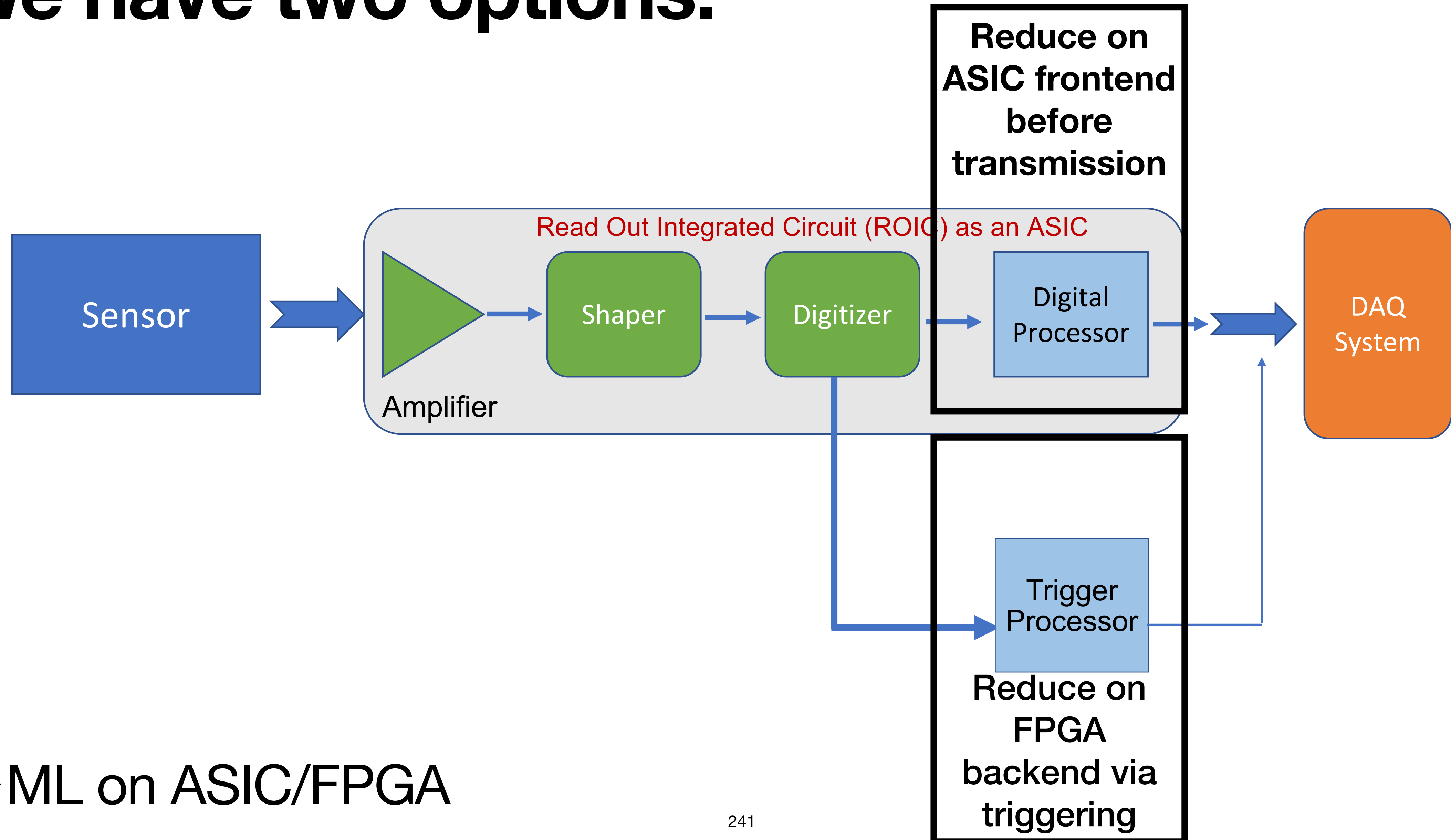
Example vertex detector inner layer from before (#241 FCC-ee):

32 bits pixel data in inner layer @ 200 MHz/cm²

➡ **24.4 Gbit/s**

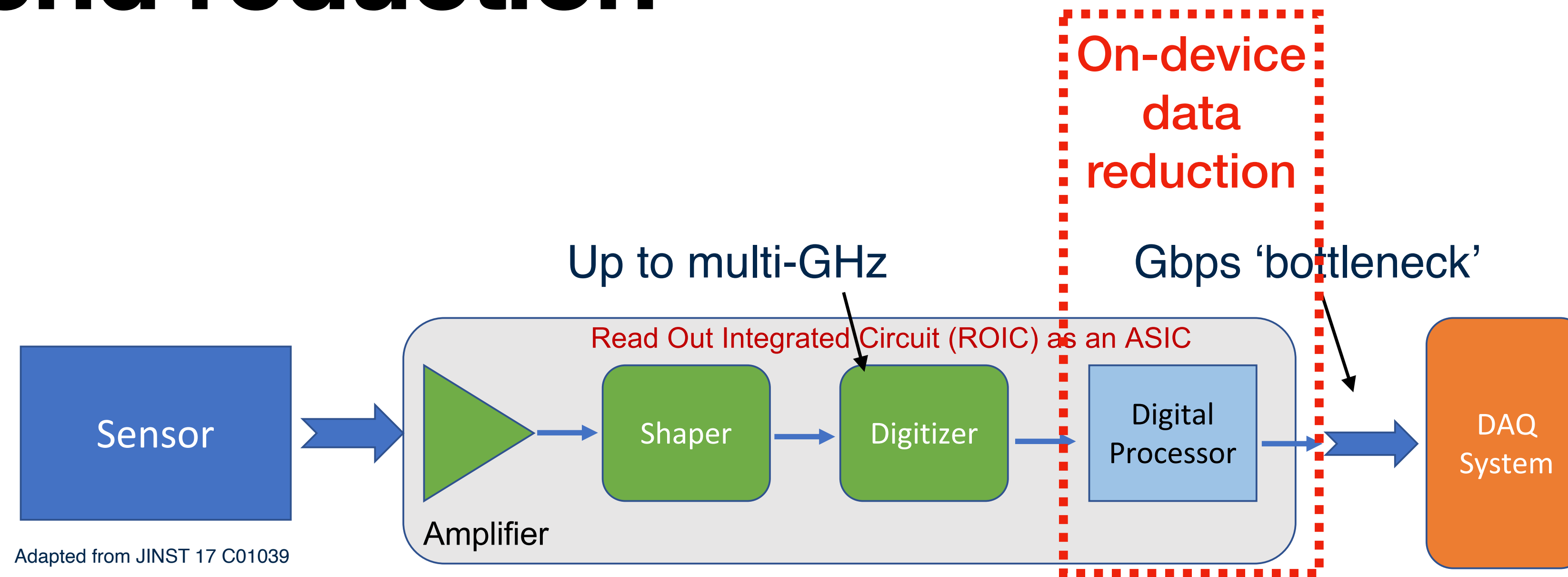
Current technology 3.2 - 6.4 Gbit/s

We have two options:



→ ML on ASIC/FPGA

Frontend reduction

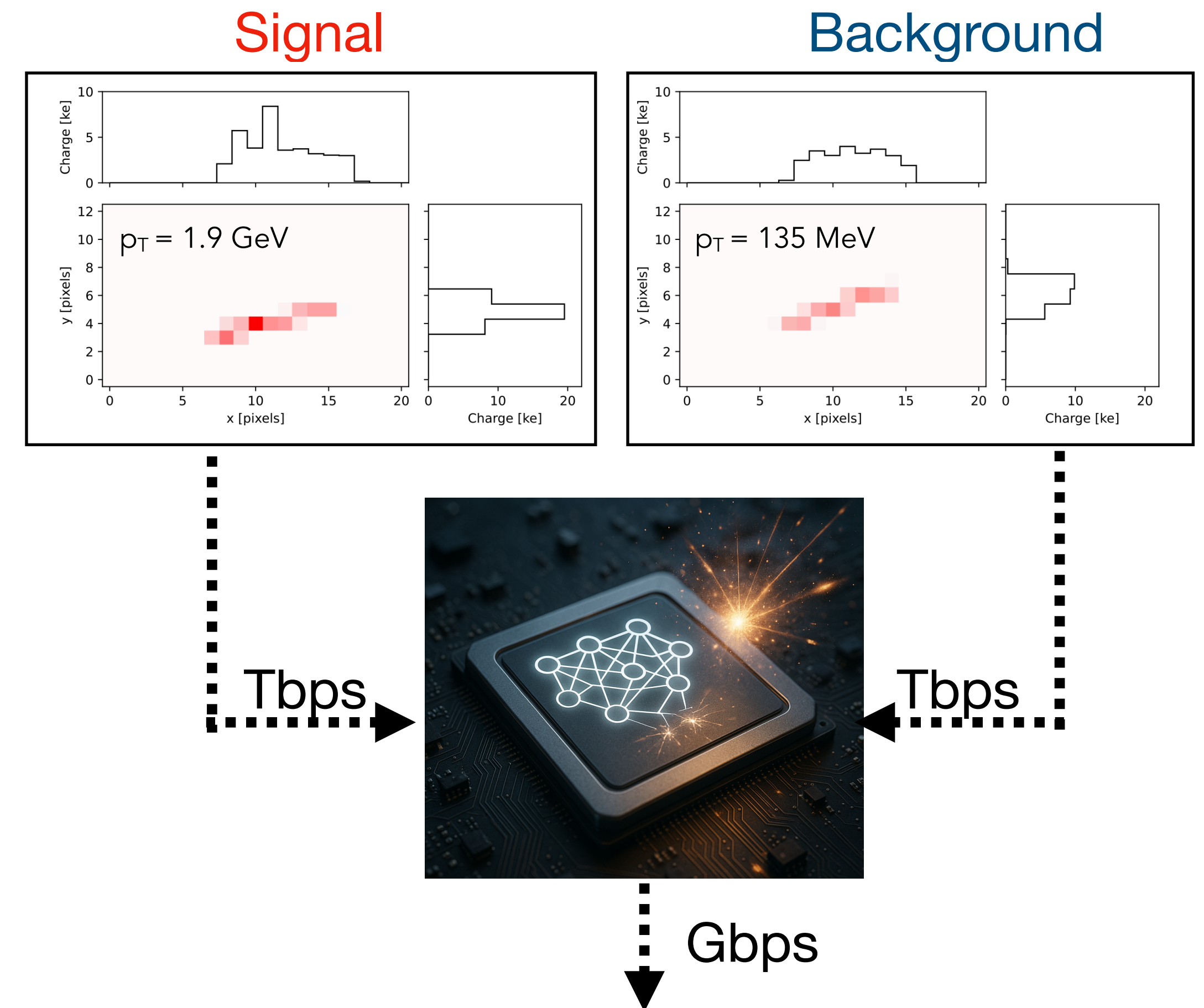


- highly granular detectors, background → high data transfer rate
- limited by **available bandwidth** of electrical and optical links
- 2 solutions: frontend reduction or **higher capacity** transfer

Smart Pixels

<https://fastmachinelearning.org/smart-pixels/>

- Reduce silicon data via **in-pixel intelligence**
 - **frontend filtering**: discard low- p_T track data (< 2 GeV)
 - **feature extraction**: Extract particle position and angle in pixel front-end ASICs from charge in **single pixel layer**
- Bandwidth savings of 57-75%!

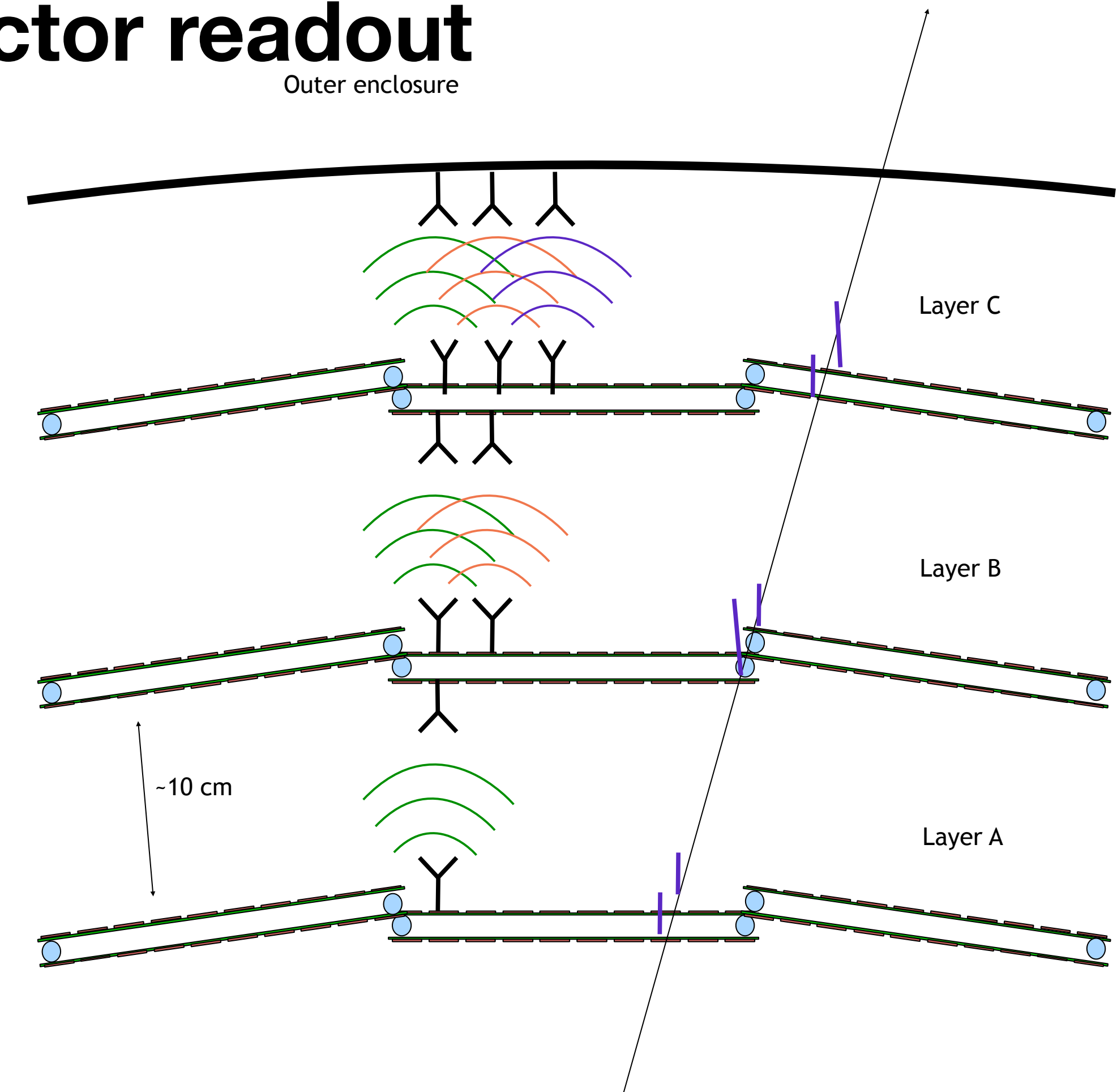


Wireless data transmission

Reducing material budget for detector readout

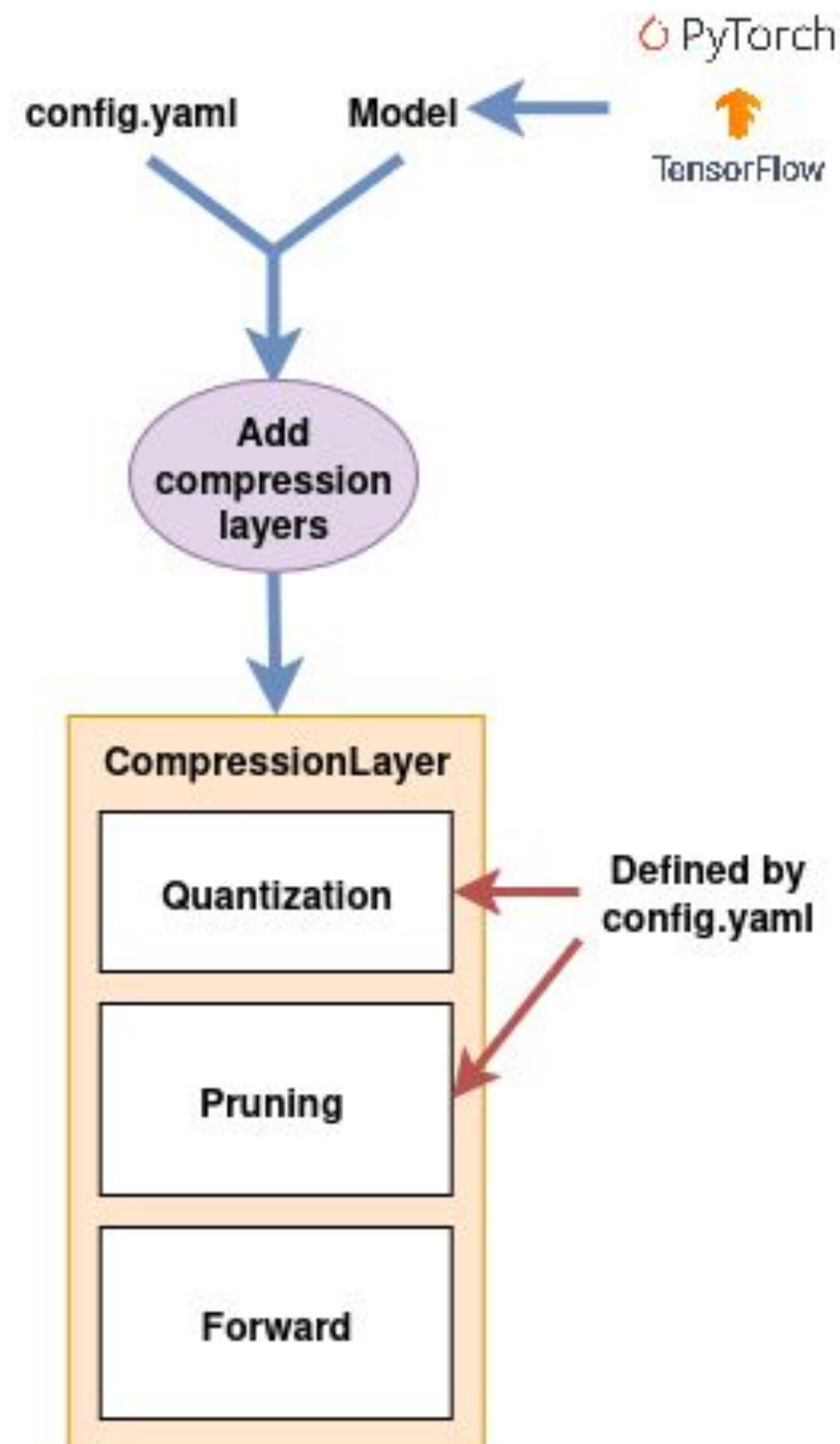
- Increase Gbps w/o increasing material
- Send single signal to several receivers, saves cabling
- Cost reduction, simplified installation/repair, reduction in dead material
 - Especially important for future tracking detectors

Few Gbps possible with 802.11ac/ad WiFi!



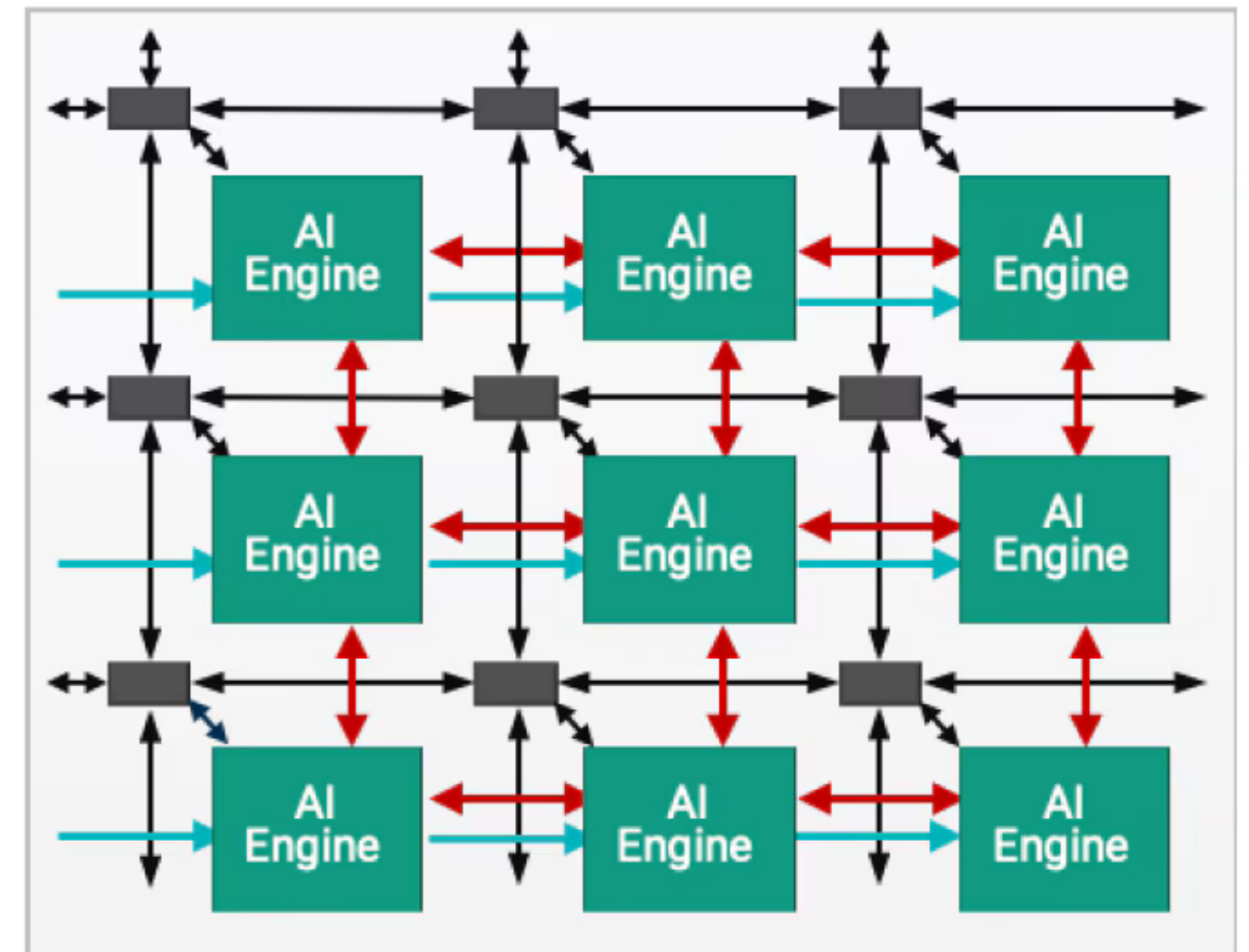
Wireless Allowing Data And Power Transmission (DRD-7.1c)

PQuantML

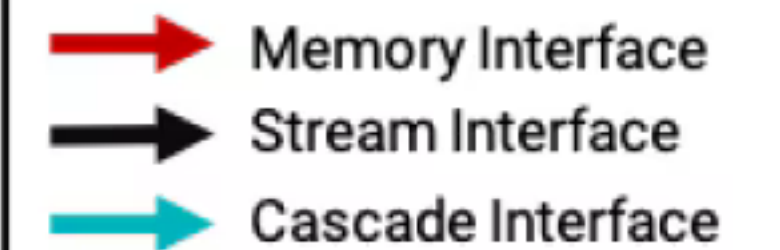


<https://pquantml.readthedocs.io/en/latest/index.html>

aie4ml



Flexible Interconnect



<https://github.com/dimdano/aie4ml>