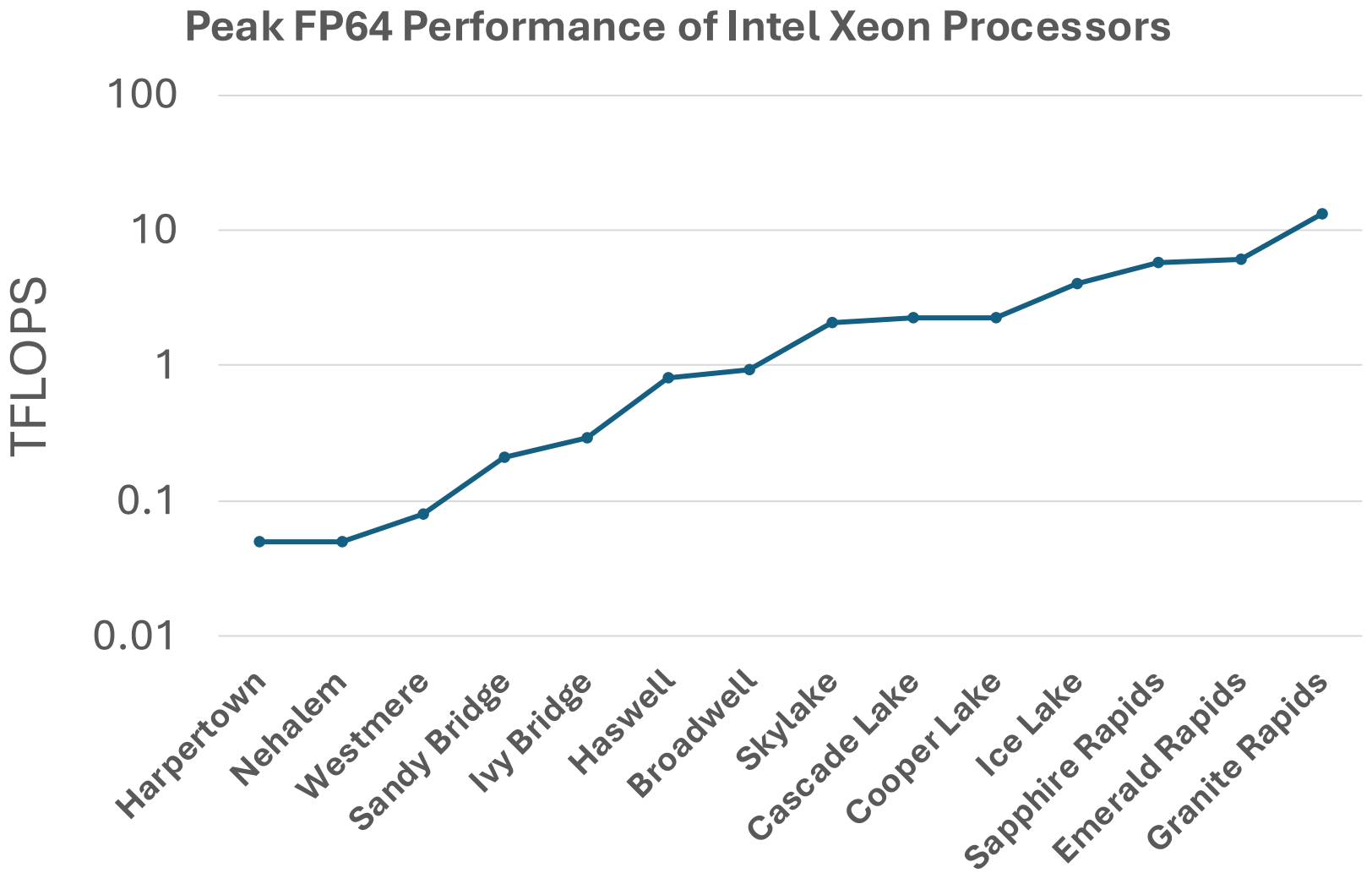
The Mixed Precision Landscape

Hardware trends, numerical methods, software, and scientific reliability

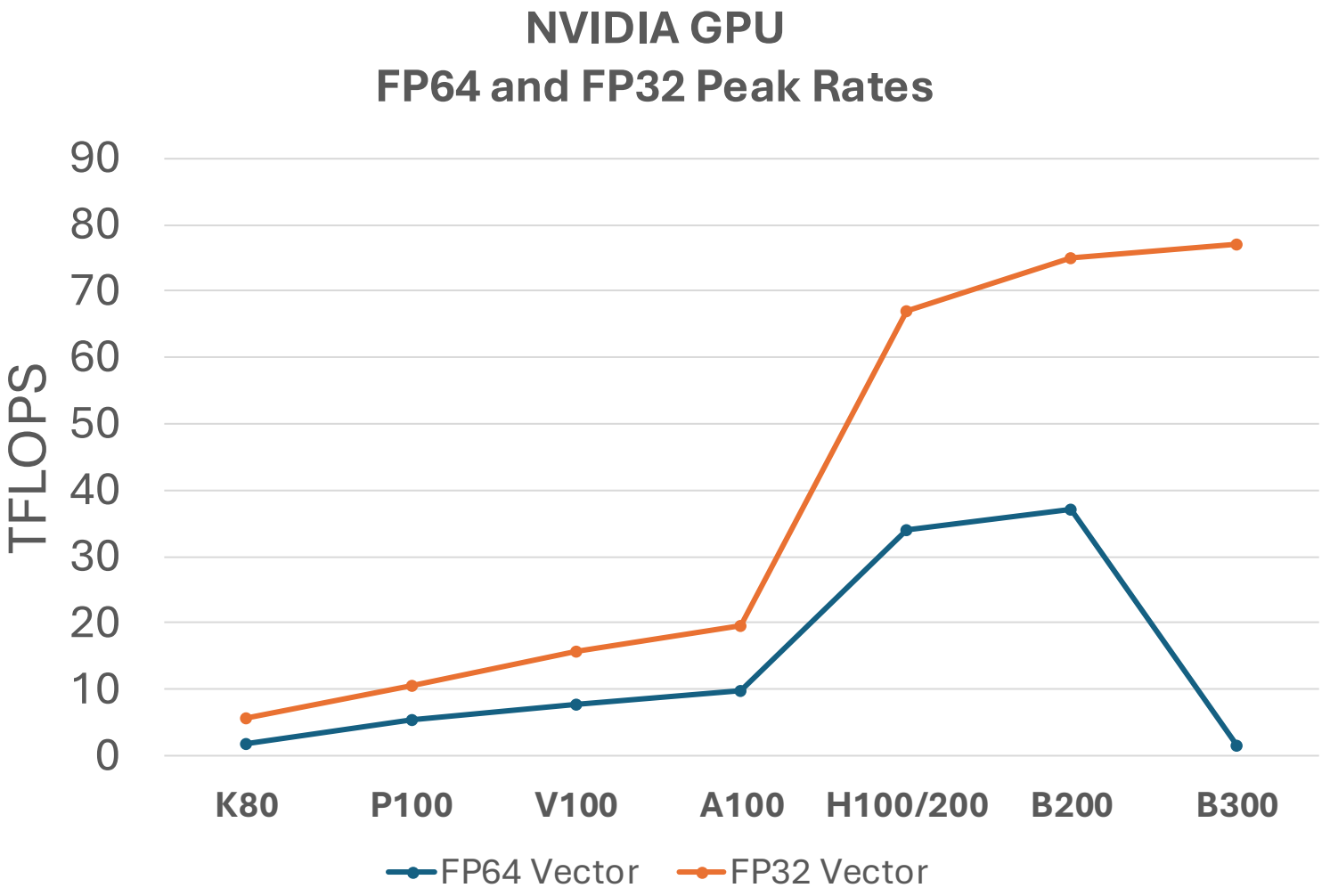
Esteban Rangel
Assistant Computational Scientist
Computational Science Division (CPS), Argonne National Laboratory

Hardware Trends Impacting 64 Bit Performance

- Historically, CPU hardware evolution reliably increased FP64 performance.

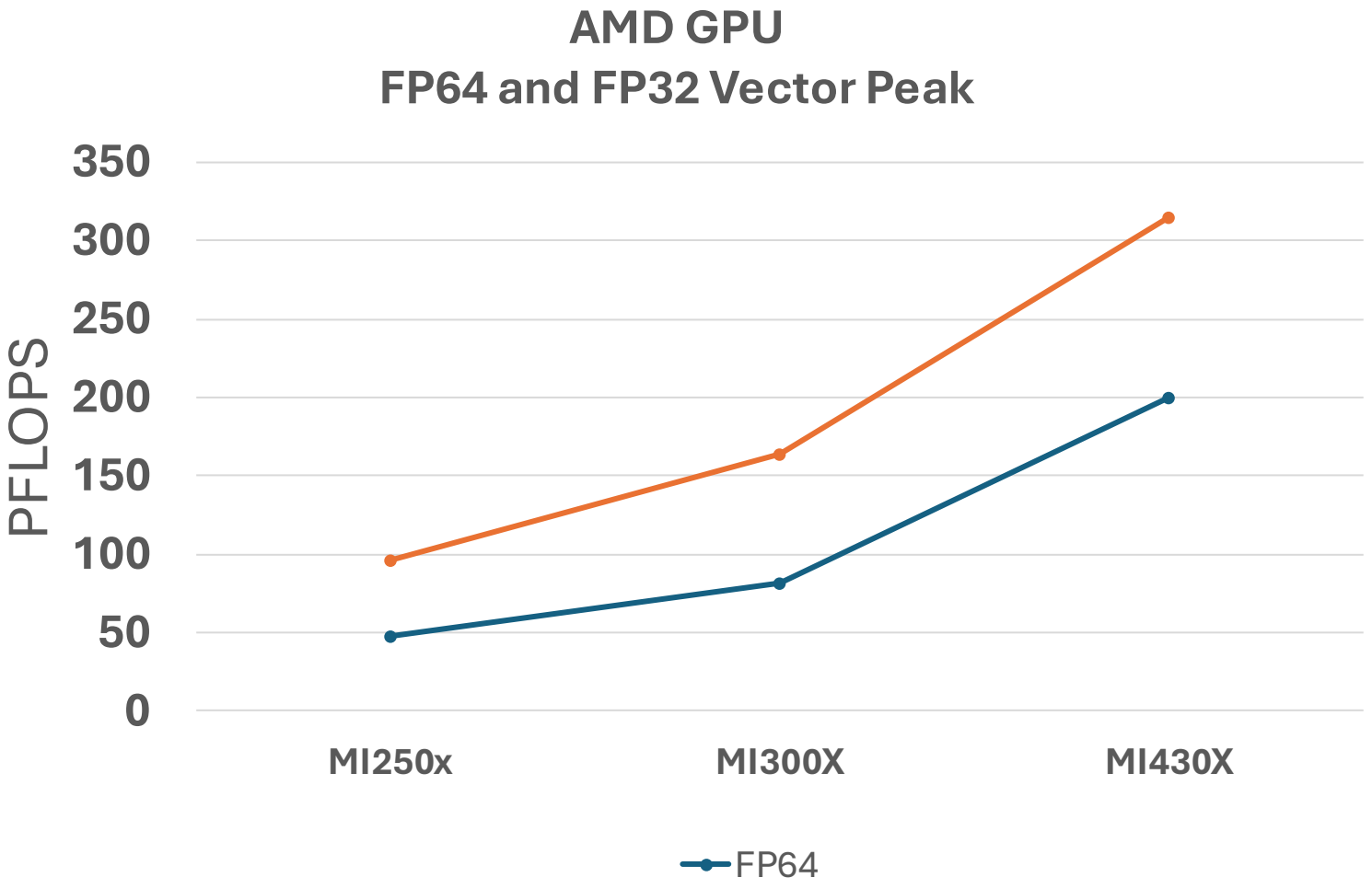


NVIDIA Vector FP32/64 Performance



- Earlier generations of NVIDIA GPUs had regular improvements in vector FP64 performance
- More recent generations have shown less consistent increases
  - Large jump with H200 (34 TFLOPS vs 10 TFLOPS on A100)
  - Smaller increase with B200 (37 TFLOPS)
- B300 targets AI and has minimal FP64 performance but continues to have improved FP32 performance

AMD Vector FP32/64 Performance

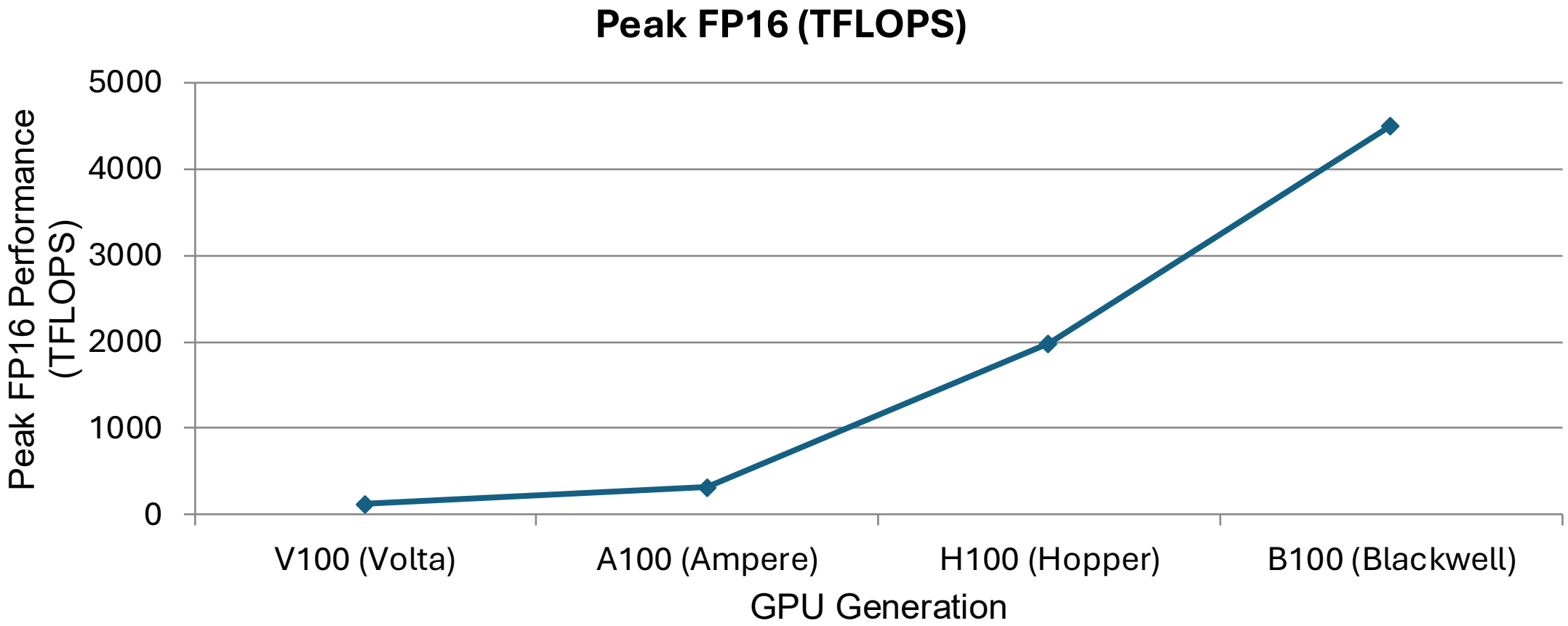


- AMD has fewer generations of hardware but continues to show increasing FP64 performance

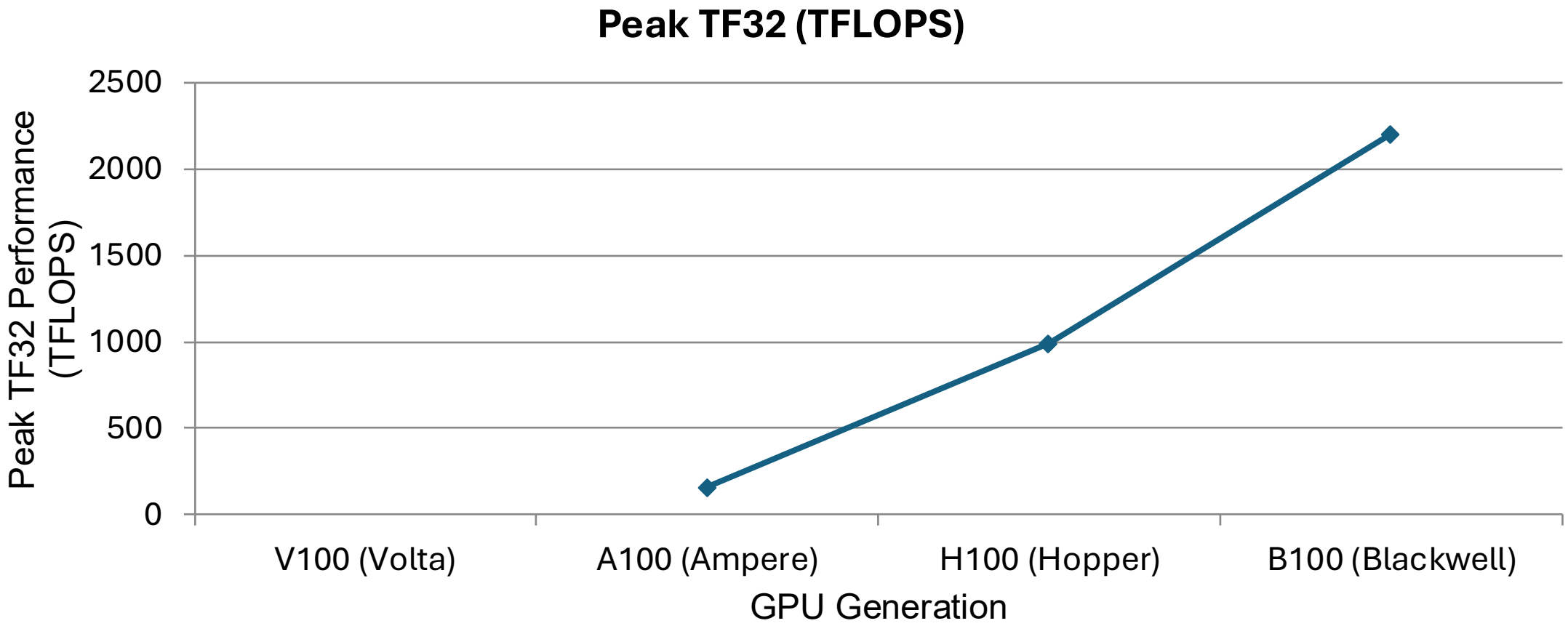
Benefits of Computing In Lower Precision

- Performance - performance in FP32 or lower precisions is generally higher than fp64 on most chips
- Hardware options — some accelerators prioritize FP32 and reduced-precision throughput over conventional FP64.
- Memory - System memory size is likely to be heavily constrained by current pricing so upcoming systems may not see significant increases in memory capacity. Using reduced precision allows for larger problems and better memory bandwidth throughput
- Power – Reduced precisions calculations can generally be performed at lower energy cost, reducing power consumption and thermal throttling.

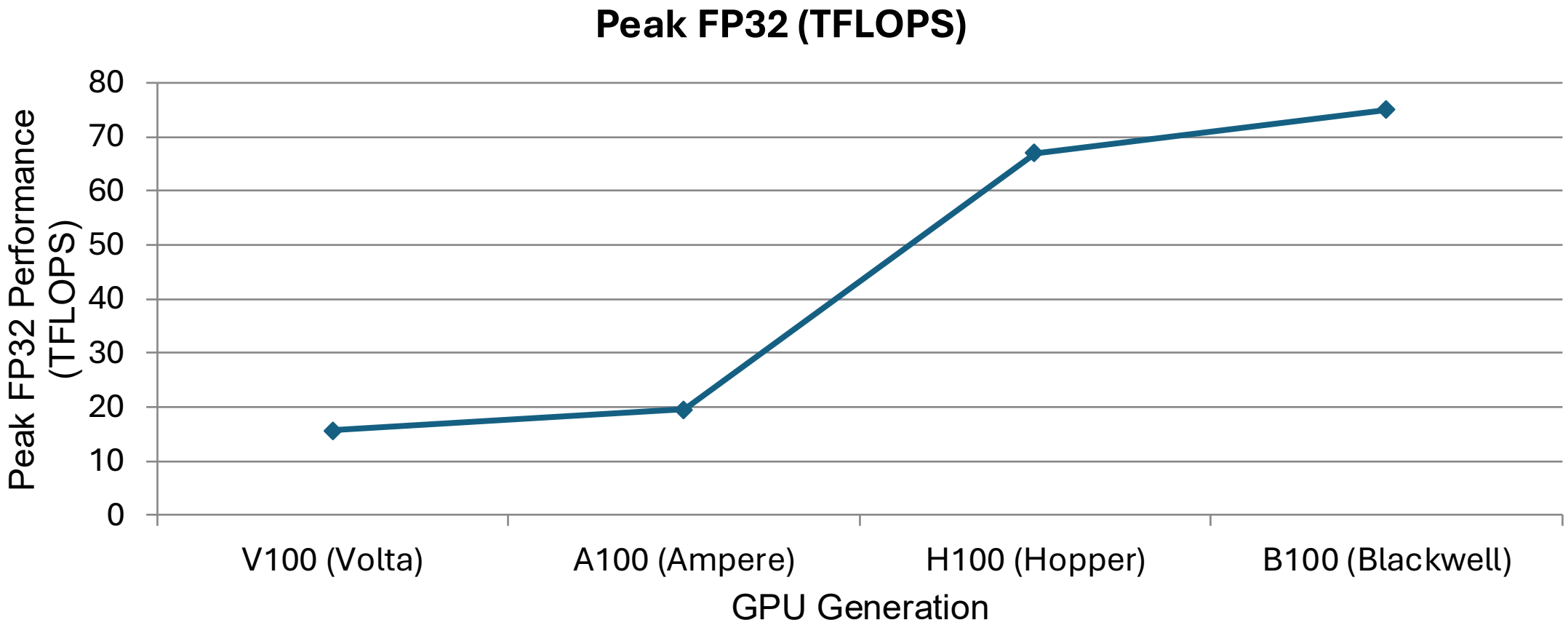
Precision Throughput Is Diverging



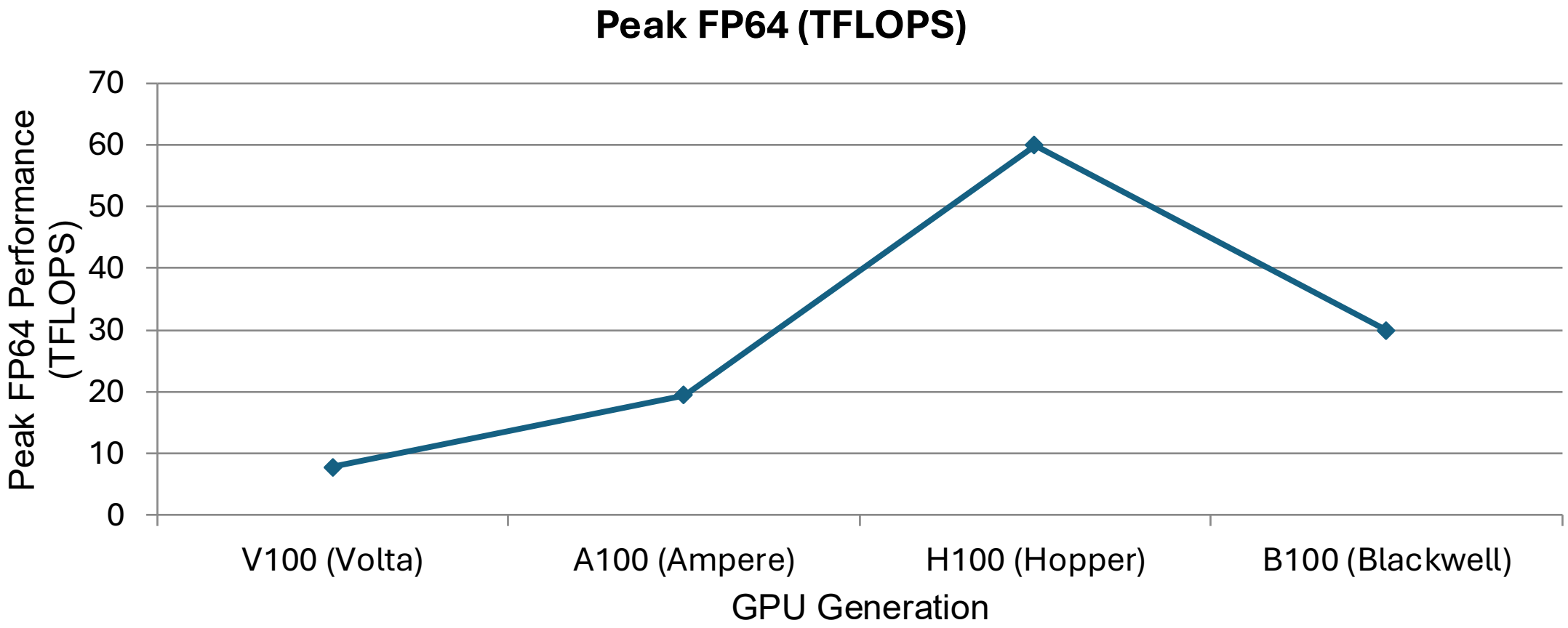
Precision Throughput Is Diverging



Precision Throughput Is Diverging



Precision Throughput Is Diverging



Nvidia B200 Performance

Floating-point format	Peak performance (TF/s)
FP64	37
FP32	75
TF32	2200
FP16	4500
FP8	9000
FP4	18000

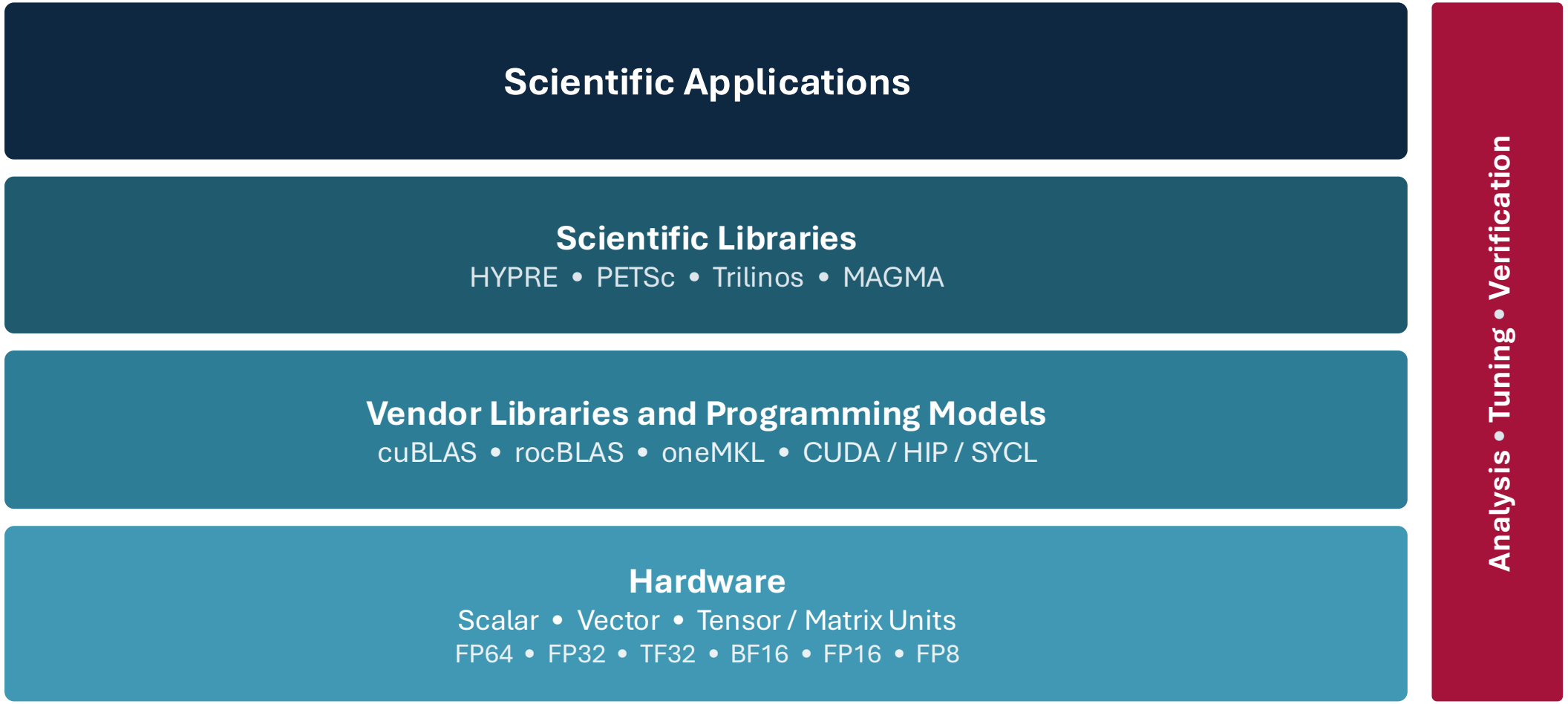
Consequences and Choices for Scientific Applications

Preserve	Selectively Convert	Redesign
Existing FP64 algorithms	Lower precision in suitable regions	Mixed-precision numerical methods
Lowest development risk	Requires sensitivity analysis	Requires algorithmic expertise
May leave performance unused	Often best first step	Potentially exposes tensor hardware
Maximum continuity	Mixed storage and compute	Iterative refinement, emulation, adaptation

For each application component:

- What accuracy is actually required?
- Is it compute-, memory-, or communication-bound?
- Can it use matrix hardware?
- What precision is used for accumulation?
- How will correctness be validated?

The Software Ecosystem Is Responding



What Does Mixed Precision Look Like in Practice?

Precision can differ across storage, operands, arithmetic, accumulation, and validation.

Lower-precision inner solve

Perform the expensive solve or preconditioner in lower precision; evaluate residuals and corrections in higher precision.

Mixed compute and accumulation

Use FP16, BF16, TF32, or FP8 operands while accumulating into FP32 or FP64.

TARGET

High-quality scientific result

Selective precision

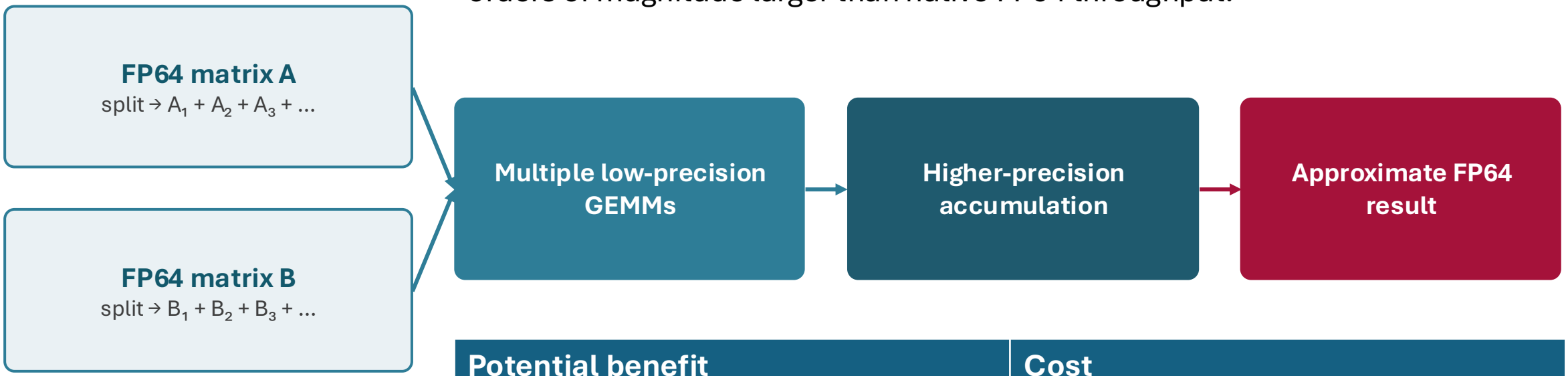
Keep sensitive kernels, reductions, convergence tests, or state variables in higher precision.

Adaptive precision

Begin cheaply and increase precision only when convergence or error criteria require it.

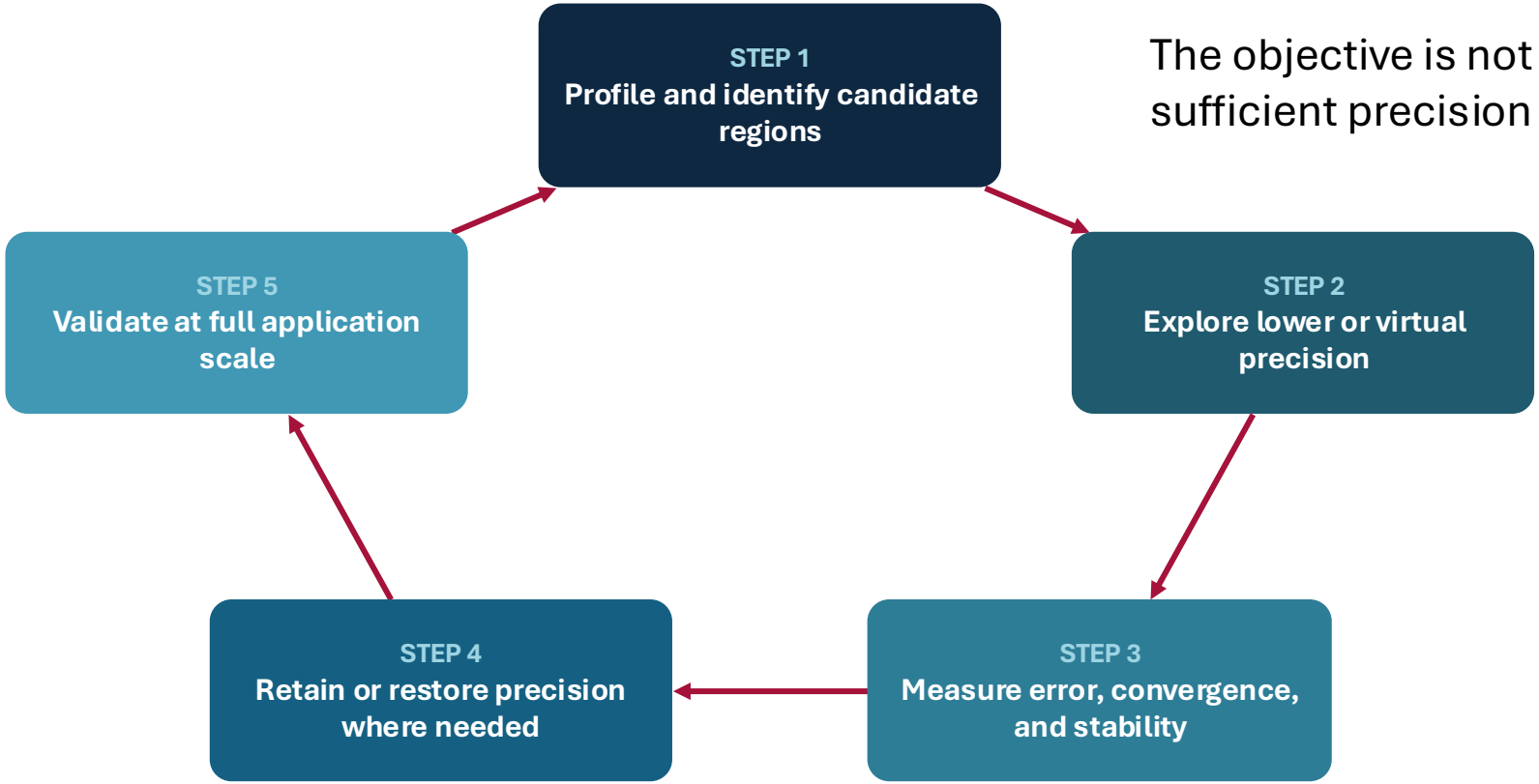
Precision Emulation on Low-Precision Hardware

A higher operation count can still win when low-precision matrix throughput is orders of magnitude larger than native FP64 throughput.



Potential benefit	Cost
Access tensor engines	More GEMMs
Higher effective throughput	Extra data movement
Portable high-accuracy strategy	Accuracy depends on decomposition
Useful when native FP64 is weak	Not beneficial for every problem size

Analysis, Experimentation, and Verification



The objective is not minimum precision everywhere. It is sufficient precision at every stage of the computation.

KEY IDEA

Precision selection is an iterative engineering process.

Exploration	Validation
VPREC, stochastic arithmetic, compiler tools	References, invariants, convergence, regression tests

The Frontier Is Moving Quickly

Precision frontier

From native FP64 toward algorithmic reconstruction

- FP8 and narrower tensor formats are receiving most of the throughput growth.
- Recent work asks whether FP64-quality results can be reconstructed using low-precision matrix engines.
- Ozaki-style methods treat low precision as an implementation substrate rather than the final accuracy target.
- The relevant question becomes: What numerical and software scaffolding is required?

Architecture frontier

Beyond conventional CPUs and GPUs

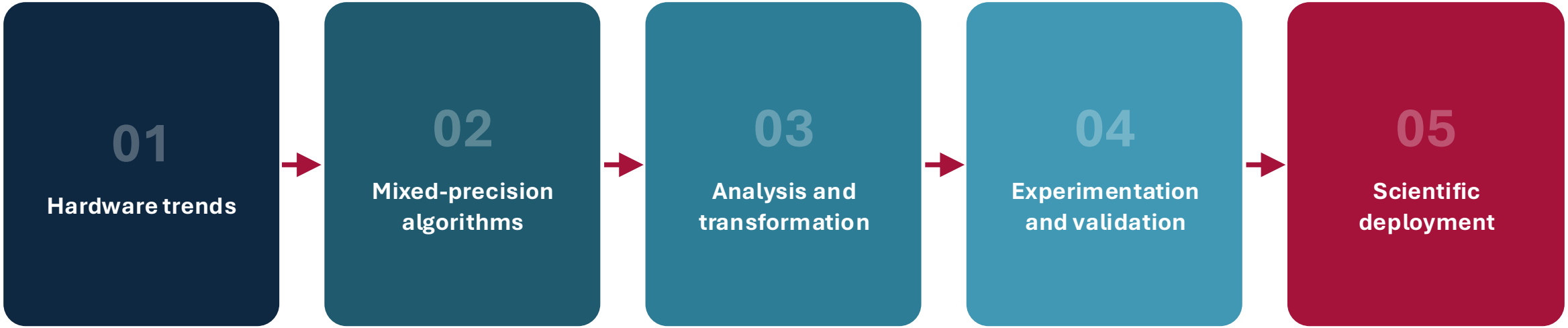
- Sandia's *Spectra* system uses NextSilicon Maverick-2 runtime-reconfigurable accelerators.
- The ALCF AI Testbed exposes wafer-scale, dataflow, and other specialized accelerators.
- These systems may provide different arithmetic features, execution models, and software interfaces.
- Developers may need libraries and tools that abstract more than vendor-specific instructions.

We've been here before

- Computers without hardware floating point units (e.g., 70s and 80s workstations) relied on software emulation; double precision typically 10x-20x slower than single precision
  - Led to careful numerical analysis of conditions under which computations could be safely performed in single precision
  - Led to new algorithms using a mix of single and double precision
  - Led to methods for estimating roundoff error
- Early general-purpose GPUs (GPGPUs) in the mid-late 2000s typically offered 8-16x single precision performance compared to double precision
  - Led to adaptation of old algorithms to this new setting and the development of mixed-precision versions of newer algorithms

From Landscape to Practice

Goal for the session: understand not only how to use lower precision, but how to use it deliberately and reliably.



ARGONNE TRAINING PROGRAM ON EXTREME-SCALE COMPUTING

Produced by Argonne National Laboratory, a U.S. Department of Energy Laboratory managed by UChicagoArgonne, LLC under contract DE-AC02-06CH11357.

Special thanks to the National Energy Research Scientific Computing Center (NERSC) and Oak Ridge Leadership Computing Facility (OLCF) for the use of their resources during the training event.