

# HPC-AI CONVERGENCE

CO-CHAIRS: MOHAMED WAHIB (RIKEN), EMMANUEL JEANNOT (INRIA),  
ARTUR LORENZON (UFRGS), PHILIPPE NAVAUX (UFRGS), BRIAN SPEARS  
(LLNL)

# ORGANIZATION

- **3 talks:**
  - Sovereignty(Emmanuel Jeannot)
  - Scalability (Artur Lorenzon)
  - AI-HPC Divergence (Mohamed Wahib)
- **Breakout sessions** by small groups (on each specific subtopics (for 1 h):
  - design: Algorithms + architectures for transformers, foundation models?
  - Scalability: Memory limits, heterogeneous arch, slow interconnects, low-precision?
  - Distributed learning: Train on inherently distributed data (sim traces)?
  - Sovereignty: Build-vs-buy for European AI-HPC stack?
  - Data: Access, open models vs open data, PB-scale science datasets?
- **30 minutes wrap-up.**

# HPC-AI CONVERGENCE: 1<sup>ST</sup> SESSION

## Sovereignty

- Goal: autonomy without isolation (avoid lock-in across compute, data, software, models, ops).
- Key levers: control hardware, data location/flows, software stack, model training, infra and skills.
- Europe gap: chips + cloud → needs long-term focused investment; be contributor, not just user.
- Data issues: curation, multilingual gaps, embedding in models complicates ownership/trust.
- Open source helps avoid silos; cost trade-off cloud vs on-prem (utilization critical).
- Now a technical HPC issue: access to large-scale AI compute (AI Factories, etc.).

## Scalability

- Core bottlenecks: mixed precision, memory limits, interconnect, hardware heterogeneity.
- Programming complexity rising (many accelerators); need better abstractions for data locality.
- AI hardware dominates → HPC must adapt workflows and codes.
- FP64 declining on new chips; emulation/low-precision (e.g., INT8) trade-offs + validation needed.
- Performance tuning: rewrite apps, use profiling to choose precision.
- Network programmability can help but is hard to use.

# HPC-AI CONVERGENCE AND DIVERGENCE

- Convergence vs divergence across hardware, software, and workloads
- Hybrid architectures combining HPC and AI systems
- Virtualization, containers, elasticity, and multi-tenancy
- Micro-heterogeneity and disaggregated inference
- Integration challenges (storage APIs, cloud/HPC bridging)
- Future aspects (quantum integration, latency constraints)

# SOVEREIGNTY IN HPC-AI

- Strategic autonomy (compute, data, software, models, operations)
- Freedom of choice vs technological lock-in
- Geopolitical and economic motivations
- Need to be contributors, not just users
- Role of open source in avoiding silos

# FIVE LAYERS OF SOVEREIGNTY

- Compute sovereignty (hardware control, alternatives to dominant vendors)
- Data sovereignty (location, access, curation, language coverage)
- Software sovereignty (portability, auditability, ecosystem control)
- Model sovereignty (training, adaptation, diversity of models)
- Operational sovereignty (infrastructure, cloud, expertise, business models)

But also:

- Skills (we need competent people to not not be lock and dependent)

# SOVEREIGNTY AS A TECHNICAL RESEARCH CHALLENGE

- Dependence of frontier AI on large-scale HPC infrastructure
- Sovereignty as access to AI-capable systems
- European initiatives (AI Factories, AI GigaFactories)
- Shift from geopolitical issue to HPC research problem

# DATA SOVEREIGNTY AND TRUST

- Data embedded in models (complexity of control and ownership)
- Trade-off between collaboration and protection
- Trust, privacy, and data leakage concerns
- Role of HPC infrastructure in addressing societal challenges
- Multi-duplication to avoid bottleneck in access

# SCALABILITY CHALLENGES IN HPC-AI SYSTEMS

- Hardware heterogeneity (CPUs, GPUs, AI accelerators, etc.)
- Memory limitations
- Interconnect bottlenecks and programmability
- Difficulty of programming complex systems

# MIXED PRECISION AND NUMERICAL ACCURACY

- SHIFT FROM FP64 TO LOWER PRECISION FORMATS
- Impact ON SCIENTIFIC APPLICATIONS (BIOINFORMATICS, OIL & GAS, ETC.)
- SELECTIVE AND ADAPTIVE PRECISION STRATEGIES
- VALIDATION AGAINST FP64 BASELINES
- CO-DESIGN BETWEEN HARDWARE, ALGORITHMS, AND APPLICATIONS

# RESOURCE UTILIZATION AND SCHEDULING

- Mapping workloads to the most suitable hardware
- Kernel-level profiling (compute, memory, communication patterns)
- Data-driven scheduling using telemetry and traces
- Dynamic runtime systems for adaptive execution
- Trade-offs: performance, energy, accuracy, throughput

# HARDWARE–SOFTWARE CO-DESIGN AND MARKET DYNAMICS

- AI-driven hardware evolution dominating HPC needs
- Limited availability of non-AI hardware
- Need to adapt HPC workflows to AI-oriented systems
- Validation challenges when porting applications
- Influence of hardware choices on software design

# COST MODELS AND INFRASTRUCTURE STRATEGY

- Cloud vs on-premise trade-offs
- Importance of high utilization (efficiency, token-based economics)
- Lack of cost optimization today (focus on performance first)
- Long-term Total Cost of Ownership (TCO) considerations

**HPC-AI CONVERGENCE / DIVERGENCE**

**THANKS!**