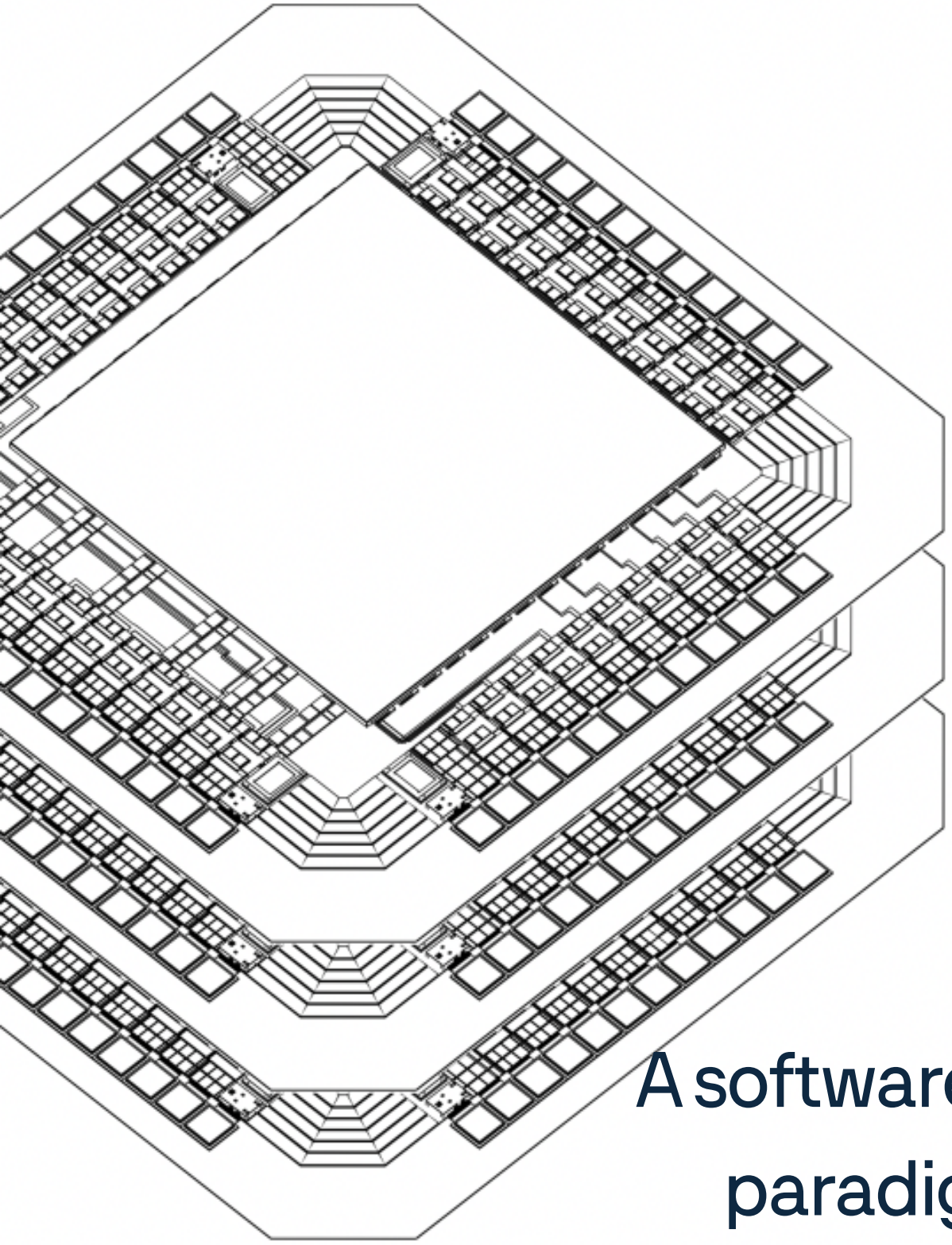


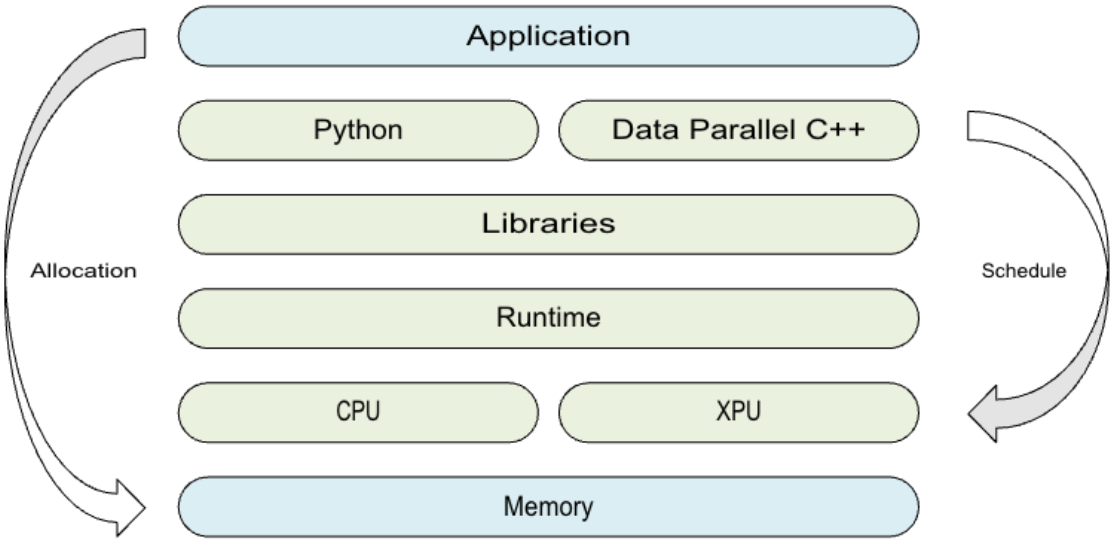
A software-first paradigm for computing architectures

VAINE

Introduction: From HW/SW to SW/HW Codesign

The relationship between hardware and software has traditionally been one of constraint: hardware capabilities define boundaries within which software must operate. This paradigm often slows developer velocity, as teams must dedicate significant time and resources to hardware-specific optimization rather than focusing on innovation. When developers are constrained by fixed hardware limitations, the result is not only slower development cycles but also systems that fail to reach their full potential for performance and efficiency.

What's typically called "hardware/software codesign" is, in practice, a process where software adapts to predetermined hardware constraints. Vaire inverts this relationship through true "software/hardware codesign"—a process where software requirements drive hardware decisions. This subtle but powerful reordering of priorities places software at the center of system design.



Vaire's Software-First and Memory-Centric Architecture

Our software-first paradigm begins by understanding computational challenges from the user's perspective. Rather than forcing software to adapt to predefined hardware constraints, we develop adaptable, high-level software that clearly expresses computational needs. Hardware is then designed specifically to meet the performance requirements articulated by this software. This approach creates systems that are inherently more efficient, adaptable, and economically viable.

The Vaire platform is built upon three foundational pillars:

- **Ease of Use**: By employing a unified address space that abstracts hardware complexity, developers can focus on solving problems rather than managing hardware limitations.
- **Energy Efficiency**: Through advanced memory solutions, adiabatic reversible computing, and innovative cooling strategies, Vaire maximizes performance while minimizing power consumption.
- **Economic Scalability**: Our streamlined development process significantly improves developer velocity, reducing both time-to-market and total ownership costs. This allows organizations to allocate more resources to innovation rather than hardware adaptation.

Core Architecture: The Unified Address Space

At the foundation of Vaire's approach is the unified address space—a concept that treats an entire rack as a single, coherent compute unit. This distributed memory architecture optimizes data movement by creating a memory-centric design where compute cores become interchangeable resources within the system.

The unified address space eliminates traditional hardware boundaries, allowing applications to access computational resources without concerning themselves with physical location. This abstraction creates three immediate benefits:

1. **Simplified Deployment**: Applications can be deployed across varied hardware configurations with minimal reconfiguration.
2. **Optimized Data Flow**: By prioritizing memory access patterns over processor arrangement, data movement—often a significant performance bottleneck—is minimized.
3. **Hardware Flexibility**: Compute resources can be added, removed, or reconfigured without requiring extensive application redesign.

This foundation enables our three-phase design methodology:

1. **Analysis of Application Challenges**: We begin by assessing specific computational and data movement requirements.
2. **Development of Adaptive Software**: We create high-level software solutions that effectively express these requirements.

3. **Implementation of Tailored Hardware**: Finally, we engineer hardware components that meet the performance metrics derived from software requirements.

The Vaire Software Stack: Bridging Abstraction and Performance

Vaire's software stack balances high-level abstraction with performance optimization through a multi-layered approach:

- **Core Programming Layer**: Our foundation is a variant of data-parallel C++ inspired by SYCL principles, providing both abstraction and direct hardware access when needed.
- **Application Framework**: A PyTorch-compatible interface supports both AI applications and general-purpose computing, ensuring versatility without sacrificing performance.
- **Integration Layer**: Built to incorporate emerging technologies such as Triton and the Modular Mojo Open Source Stack, our platform remains future-proof as the computational landscape evolves.
- **Hardware Acceleration**: RISC-V CPU cores work alongside AI matrix acceleration coprocessors, creating a scalable, high-performance computing environment.

The stack includes integrated visual telemetry tools similar to Perfetto trace, enabling developers to monitor and analyze data flow intuitively. These tools significantly reduce the learning curve and accelerate development cycles by providing clear visibility into system behavior.

Advanced Technologies: Maximizing Efficiency and Density

In today's environment, where data-intensive workloads and tight energy constraints challenge traditional architectures, pushing the boundaries of efficiency and compute density is essential. Building on the robust capabilities of our software stack—a unified programming model and actionable visualizations that simplify development and optimize performance—we now turn our focus to hardware innovations.

At the forefront of our hardware strategy is adiabatic reversible computing. By recapturing energy typically lost in conventional logic circuits, this approach not only delivers significant power efficiency improvements but also unlocks a new perspective on system investments. It enables a more balanced allocation of resources between compute performance and the physical design of high-density environments.

Complementing this innovation, Vaire leverages a modular architecture where compute blocks—each equipped with high-speed local DRAM and proportional compute resources—offer the specialized performance of dedicated accelerators alongside the flexibility of general-purpose computing. Advanced thermal management techniques further ensure that these densely packed systems operate optimally, collectively redefining how high-performance systems are built and deployed.

Adiabatic Reversible Computing

Adiabatic Reversible Computing (ARC) is a specialized logic family particularly well suited to computational tasks. Unlike conventional CMOS logic that dissipates energy with each state transition, ARC circuits recover and reuse energy by following reversible logic principles. This approach trades increased silicon area for significant energy recovery, making it ideal for reducing power consumption in compute-intensive workloads.

Vaire implements a hybrid approach at the silicon level, applying reversible computing where it delivers maximum benefit while maintaining software compatibility with existing programming models:

- **Data Plane Operations**: High-intensity, repetitive computational tasks like matrix multiplications and data transformations leverage reversible adiabatic logic. These operations see significant power efficiency improvements while preserving the same software interfaces.
- **Control Logic Operations**: Traditional CMOS logic handles control plane operations, low-latency tasks, and complex decision-making processes where reversible computing would be less advantageous.

By selectively implementing reversible logic gates and adiabatic switching, Vaire improves energy efficiency without requiring developers to learn new paradigms. This approach bridges theoretical energy conservation principles with practical computing needs, all while building upon familiar software contracts and established programming models.

Memory-Centric Hybrid Architecture

By prioritizing memory access patterns over traditional processor arrangements, Vaire minimizes data movement—often a significant bottleneck in modern computing. Our architecture represents a hybrid approach that combines two complementary resource types within a unified programming model:

1. **High-Performance Compute Subsystems**: These consist of specialized high-bandwidth memory directly attached to proportional units of compute acceleration and network scale-up bandwidth. These subsystems excel at data-intensive operations that benefit from tight memory-compute integration.
2. **Commodity CPU Resources**: Standard processors paired with high-capacity DRAM provide flexible, general-purpose computation for control flow and less specialized workloads.

The key innovation lies in how these disparate resources are unified at a fundamental level. Vaire's software stack treats all compute resources as interchangeable elements within the system, abstracting away the underlying hardware differences. This unification enables:

- **Resource-Aware Kernel Execution**: The architecture reduces the cost of experimenting with different compute schedules across heterogeneous resources.
- **Transparent Data Placement**: Data can be placed in either high-bandwidth specialized memory or high-capacity commodity memory based on access patterns and computational requirements.
- **Dynamic Resource Selection**: The system can heuristically determine whether a workload would benefit more from specialized accelerators or general-purpose CPUs.

Our co-designed software tools and hardware resources empower developers to map computations to the most suitable resources, without requiring deep hardware expertise. This integrated approach delivers the high-performance advantages of specialized hardware while preserving the adaptability of general-purpose computing, all without burdening developers with the complexities of managing a heterogeneous system.

Advanced Thermal Management

The industry is rapidly moving toward higher-density computing, with current data centers deploying 120kW racks and roadmaps targeting up to 600kW per rack. This trend is

enabled by liquid cooling and other advanced thermal management techniques that address the limitations of traditional air cooling.

Vaire builds upon and extends this industry direction through rack-level thermal design optimization. Our integrated approach to cooling:

- **Power Envelope Maximization**: Our rack-level cooling architecture allows more computational resources to operate within the fixed power constraints that typically limit data center deployment.
- **Component Protection**: Closed-loop cooling systems shield sensitive hardware from particulates and environmental contaminants, making Vaire systems particularly suitable for deployment in harsh environments.
- **Energy Recovery Integration**: Future generations will feature progressively tighter integration between cooling and compute systems, recovering energy that would otherwise be lost as heat and enabling even higher component density.

These thermal management strategies, combined with our memory-centric design and energy-efficient computing approaches, create systems that are more powerful and cost-effective than traditional architectures. Vaire's approach aligns with broader industry trends toward higher density computing while preserving the software abstractions that developers rely on for productivity.

Real-World Applications: From Concept to Implementation

Vaire's North Star for rack-level computing is that developers can deploy at-scale workloads with minimal code. With perhaps as few as 100 lines, the full computational capacity of an entire rack becomes accessible. This approach delivers rapid results with balanced performance, streamlines development, and enables seamless integration and reuse of code across diverse applications.

Rapid Results with Balanced Performance

Vaire's objective is to deliver Pareto-level performance from initial deployment with minimal configuration effort. This approach eliminates the exhaustive optimization cycles typically required by traditional systems, while still supporting long-tail investments for organizations seeking to extract maximum performance.

Simplified Development Model

The unified address space simplifies the software contract between developers and hardware. By providing a consistent memory model across heterogeneous compute resources, developers can focus on algorithms and data structures rather than hardware-specific memory management. This abstraction eliminates many of the conceptual hurdles that typically slow development on complex computing systems.

Workload Integration and Code Reuse

While AI applications represent a primary use case, Vaire's architecture integrates non-AI control systems and general-purpose computing workloads. This versatility enables organizations to consolidate previously siloed computational resources.

Vaire's support for Data Parallel C++ and Python compatibility enables organizations to leverage and adapt existing codebases. This approach allows for incremental acceleration, starting with modest performance gains that grow substantially over time as workloads are further optimized for memory-centric computation.

The programming model encourages developers to think in terms of data and memory first, rather than specific compute resources. By focusing on data structures and memory access patterns, developers can create workloads that partition data effectively, allocate memory strategically, and determine appropriate compute schedules.

Case Studies: Vaire in Action

The following scenarios represent aspirational outcomes developed as part of our "start with the press release" design process. Envisioning customer experiences that guide our development priorities.

Financial Services: Quantitative Trading Platform

A financial services firm worked with Vaire to support their quantitative trading platform. The unified address space enabled them to reduce latency by 30%. Most significantly, their development team reduced hardware-specific optimization time by 70%, dramatically improving developer velocity. This shift allowed engineers to focus on algorithmic innovations rather than hardware constraints, enabling them to pilot multiple trading strategies simultaneously.

Research Institution: Large-Scale Simulation Environment

A scientific research organization deployed Vaire to power complex climate simulations. The memory-centric architecture allowed them to scale their models to unprecedented levels of detail while reducing energy consumption by 30%. The visual telemetry tools enabled researchers without specialized hardware knowledge to optimize their simulations independently.

Critical Security Infrastructure: Autonomous Surveillance System

A security organization deploys Vaire for site-wide monitoring across hundreds of sensors and cameras. The unified architecture enables real-time edge analysis without dependence on external networks. This autonomous capability ensures continuous security monitoring during connectivity disruptions or in remote locations. Processing all data locally within a single rack maintains operational security while providing the computational power needed for advanced threat detection.

Cloud Provider: Infrastructure Modernization

A cloud services provider adopted Vaire as part of their infrastructure modernization initiative. By implementing the software-first approach across their data centers, they achieved a 30% reduction in deployment time for new services while improving resource utilization by 20%. The economic impact included both reduced operational costs and accelerated time-to-market for new offerings.

Autonomous Systems: Edge Computing

The same software investment that powers rack-scale computing also scales down to edge devices. Vaire's unified software model enables consistent workloads across both central infrastructure and autonomous edge systems. This approach is particularly valuable for power-constrained devices requiring real-time processing for sensor fusion and decision-making.

Conclusion: The Path Forward

Vaire's software-first approach offers a shift in computer architecture by aligning hardware design with evolving software needs. Our unified programming model, intelligent

data mapping, and advanced telemetry tools create systems that are more agile, efficient, and future-proof than traditional approaches allow.

At the hardware level, our innovations in adiabatic reversible computing (ARC) enable us to invest in a truly software first paradigm. ARC recaptures energy typically lost in conventional circuits, paving the way for new levels of compute density and efficiency. This approach is further enhanced by our modular compute resources that combine high speed local DRAM with scalable compute elements, as well as by advanced thermal management techniques. Together, these innovations rebalance system investments to optimize both performance and energy consumption.

We welcome stakeholders from across the computational spectrum to explore this innovative approach. Whether your focus is high-performance computing, AI acceleration, or efficient general-purpose processing, Vaire's software-first paradigm offers a robust foundation for sustainable innovation in an increasingly complex computational landscape.