From Flow to Pieces – and Back Again
The Hidden Pattern Behind Books, Bits, Brains, and the Future of Computing

Lev Neymotin, PhD
Uncompressed Thinking Exploration

We learned to command nature by replacing it with symbols.
Now the cost of those symbols is bringing us back to nature.

If you walk into a synagogue, a museum, or an archive and see an ancient scroll, something feels immediately different. You don't flip to page 173. You unroll time itself. To reach the middle, you must pass everything that came before. Knowledge is not indexed – it is traversed.

Then, at some point in history, someone did something deceptively simple: instead of rolling the text, they folded it. Sheets were stacked, stitched, and bound along one edge. Suddenly you didn't have to travel through the past to reach the present. You could jump. Any place. Any idea. Knowledge stopped behaving like a road and started behaving like a map. With that quiet mechanical change, random access was born – and with it the modern book, scholarship, and eventually the way we organize information itself.

That change was not just about convenience. It was a transformation from a flowing continuum into discrete pieces you could point to, count, copy, rearrange, and revisit at will.

Once you notice it, the same transformation seems to repeat everywhere. Continuous pitch becomes the fixed keys of a piano. A photograph's chemical gradients become pixels. Moving images turn into frames. Locations become coordinates. Sound becomes samples. Again and again, something that flows is cut into pieces that can be stored, indexed, copied, and rearranged.

I began to suspect that all these changes belonged to the same class – a recurring transformation I now think of as the **Chunking Principle**: the conversion of a flowing continuum into discrete, manipulable units. Once chunked, the world stops behaving like water and starts behaving like Lego. A violin can produce infinitely subtle pitches, sliding smoothly between notes. A piano cannot. It offers a grid of fixed keys. Yet the piano conquered concert halls, classrooms, and living rooms because its discreteness made music reproducible, teachable, and shareable.

For centuries, this strategy quietly spread across culture and technology. Whenever something continuous proved hard to control, store, or transmit, we sliced it into pieces. Eventually the process reached its logical extreme: we began representing reality itself as sequences of tiny symbols – zeros and ones. Computers did not invent the Chunking Principle; they perfected it. In a digital machine, everything becomes discrete, countable, addressable. Sound, images, motion, language, even scientific models of nature are reduced to bits. It was an astonishing triumph – and like many triumphs, it hid its costs.

Film photography captured continuous tones in chemistry. Digital cameras chopped light into pixels. Magnetic tape stored sound as a continuous signal. CDs turned it into numbers. Maps once depicted terrain as drawn shapes; GPS describes location as coordinates. Even color – a seamless experience to the eye – becomes red-green-blue values inside a computer.

Again and again, humanity seems to take something that flows and cut it into pieces.

Why?

Because pieces can be indexed. Pieces can be duplicated. Pieces can survive noise. Pieces can be assembled into larger structures like bricks into a building or Lego blocks into a castle.

A flow is beautiful. Pieces are powerful.

For roughly a century, this strategy culminated in digital computing. Reality was translated into zeros and ones – the smallest possible pieces. Everything became countable, addressable, and programmable. We built machines that could manipulate these symbols at unimaginable speed. Communication, finance, medicine, science, entertainment – entire civilizations now run on vast oceans of discrete states switching billions of times per second.

And it worked astonishingly well.

But success changes the problem.

The difficulty is that the world itself never stopped being continuous. Fluids flow, fields vary smoothly, bodies move through space without jumping from coordinate to coordinate. Digital systems approximate this continuity by taking enough tiny steps that the gaps become invisible. For decades, that approximation was good enough. Now we are beginning to encounter problems where simulating reality piece by piece demands enormous effort – energy, hardware, time – while the physical processes being modeled would unfold effortlessly on their own.

As digital systems expanded into domains once reserved for human intelligence – vision, language, reasoning – something unexpected happened. The bottleneck was no longer clever algorithms or faster chips. It was electricity, heat, cooling water, supply chains, land use, and physical infrastructure. The symbolic world began to show its physical footprint.

Abstraction, it turns out, is not weightless.

Every bit flipped inside a processor requires energy. Every stored symbol occupies matter. Every data center must shed heat into the environment. When scaled to planetary levels, digital computation looks less like pure mathematics and more like heavy industry.

Meanwhile, the most sophisticated intelligence we know – the human brain – runs on roughly the power of a dim light bulb. It perceives, learns, adapts, imagines, and coordinates a body in real time, all without a single clock cycle or line of code in the conventional sense. Its computation is inseparable from the physics of neurons, chemistry, and structure. It does not simulate reality; it participates in it.

This contrast invites a provocative question: what if our digital machines are solving the wrong problem in the right way?

Digital computing simulates continuous processes by chopping them into discrete steps. Weather models approximate fluids. Neural networks approximate perception. Physics simulations approximate nature. The approach is universal – anything describable can, in principle, be computed – but universality comes at a cost. The more closely a problem resembles real physical dynamics, the more work symbolic systems must do to imitate what matter already does naturally.

Nature does not calculate the trajectory of a falling object. It simply falls.

An electrical circuit does not numerically solve differential equations. Its voltages evolve according to those equations automatically. Light passing through a lens performs a Fourier transform without knowing the term. Quantum systems explore vast spaces of possibilities simply by existing in them.
In this sense, the universe is not merely governed by laws of physics. It is continuously executing them.

Early analog computers exploited this fact by building physical systems whose behavior mirrored mathematical problems. They were fast and elegant but fragile, difficult to program, and prone to drift. Digital machines, though less natural for continuous dynamics, were reliable and scalable. The choice was clear.

Today, however, technology has advanced to the point where that trade-off may no longer be so one-sided. Precision fabrication, feedback control, new materials, machine learning calibration, and hybrid architectures allow us to stabilize physical systems in ways earlier engineers could only dream of. Instead of replacing digital computing, these approaches can complement it, letting physics handle what physics does best while digital logic manages coordination and correction.

Quantum computing is a dramatic example. A qubit is not simply a fancy bit. It occupies a continuous state space, able to exist in superpositions that have no classical counterpart. Harnessing this richness is extraordinarily difficult, requiring extreme isolation and sophisticated control, but it demonstrates that computation can arise directly from physical laws rather than from symbolic manipulation alone.

Neuromorphic chips attempt something similar at a biological scale, implementing circuits that behave more like neurons than logic gates. Optical systems compute using interference patterns of light. In-memory architectures perform operations where data reside instead of shuttling them back and forth across energy-hungry buses. Each of these efforts reflects the same intuition: perhaps the most efficient way to compute is not to abstract away from physics but to lean into it.

At this point, the earlier pattern begins to bend.

For millennia, progress meant cutting continua into discrete pieces. Now we are exploring how to reintroduce continuity where discreteness becomes too expensive. Not because analog is nostalgic or aesthetically pleasing, but because the physical world remains continuous whether we like it or not.

Even physics itself refuses to choose sides cleanly. Quantum field theory describes reality as continuous fields whose excitations appear as discrete particles. Energy comes in quanta, yet wavefunctions evolve smoothly. Measurement produces definite outcomes from probabilistic processes. The universe seems perfectly comfortable being both analog and digital at once.

Perhaps our technologies are converging on the same hybrid state.

Digital methods will not disappear. They are unmatched for exact arithmetic, long-term storage, verification, and interoperability across global networks. But they may lose their monopoly. Instead of forcing every problem into symbolic form, future systems may distribute work across multiple substrates – electronic, optical, quantum, biochemical – each chosen for physical suitability.

In that world, computation becomes less about pushing bits faster and more about orchestrating processes that unfold naturally.

There is also a deeper shift in how information itself might be represented. Digital systems build complex structures from tiny atomic symbols. Human cognition, by contrast, operates largely on compressed patterns and concepts. We remember the story, not every syllable; the face, not every pixel. Machine learning already hints at this through embeddings and hierarchical representations. One can imagine architectures that manipulate such higher-level structures directly, reducing the distance between representation and meaning.

If so, the future of computing may feel less like programming and more like shaping dynamical systems – guiding flows rather than issuing instructions.
It is tempting to call this a return to analog computing, but that phrase misses the point. We are not going back to vacuum-tube integrators or mechanical differential analyzers. We are moving toward systems that combine the reliability of digital control with the efficiency of physical processes. The goal is not to abandon abstraction, but to stop using it where it is wasteful.

Technological history often resembles a pendulum. Early tools exploit physical phenomena directly. Later technologies abstract away from those phenomena to gain flexibility. Eventually, the cost of abstraction becomes visible, and the pendulum swings back toward deeper engagement with reality – this time armed with the knowledge gained during the abstraction phase.

Seen this way, the digital revolution was not a destination but a stage.

Energy may ultimately decide how far this stage can extend. Any intelligent system must operate within a power budget. Evolution produced brains that maximize usefulness per watt because survival demanded it. Human technology, buoyed by abundant energy, could afford inefficiency for a time. As demands escalate, efficiency returns as a central constraint. Systems that achieve more with less will outcompete those that simply scale consumption.

Artificial intelligence stands at this crossroads today. Larger models deliver impressive capabilities but at mounting cost. If performance gains require exponentially more energy, the trajectory becomes unsustainable. At that point, architectural innovation is not optional; it is inevitable.

Physics-native computation offers one path forward – not because it is philosophically appealing, but because it reduces the gap between computation and the physical processes being modeled. When simulation becomes too heavy, participation becomes attractive.

None of this diminishes the achievements of the digital age. On the contrary, digital tools made it possible to design, measure, and control complex physical systems with extraordinary precision. Without them, hybrid architectures would be unmanageable. In a sense, digital computing created the conditions for its own evolution.

The deeper lesson may be that intelligence, whether biological or artificial, is not fundamentally about symbols or circuits but about effective interaction with reality under constraints. Different substrates offer different advantages, and no single paradigm is likely to dominate indefinitely.

The universe has always been computing. We are learning, once again, how to let it.

The future of computing will not be decided by ideology – analog versus digital – but by physics. Wherever symbolic abstraction remains efficient, it will endure. Wherever it becomes too costly, matter and energy themselves will resume the work.

We are not approaching the end of the digital age, but the end of its monopoly. The next era will belong to systems that compute not by describing reality, but by embodying it.