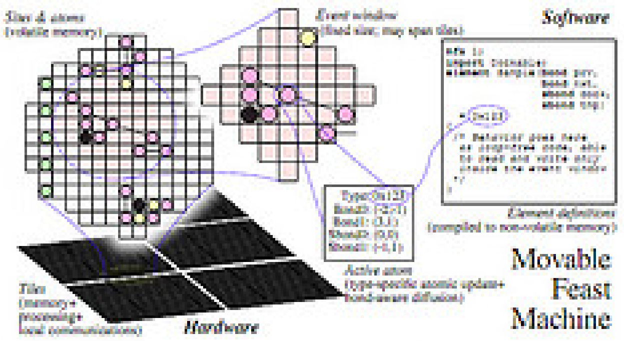
http://highscalability.com/blog/2011/9/28/pursue-robust-indefinite-scalability-with-the-movable-feast.html

Pursue robust indefinite scalability with the Movable Feast Machine

Wednesday, September 28, 2011 at 9:10AM

Todd Hoff in Paper



And now for something completely different, brought to you by David Ackley and Daniel Cannon in their playfully thought provoking paper: Pursue robust indefinite scalability, wherein they try to take a fresh look at neural networks, starting from scratch.

What is this strange thing called indefinite scalability? They sound like words that don't really go together:

> *Indefinite scalability is the property that the design can support open-ended computational growth without substantial re-engineering, in as strict as sense as can be managed. By comparison, many computer, algorithm, and network designs -- even those that address scalability -- are only finitely scalable because their scalability occurs within some finite space. For example, an in-core sorting algorithm for a 32 bit machine can only scale to billions of numbers before address space is exhausted and then that algorithm must be re-engineered.*
>
> *Our idea is to expose indefinitely scalable*

> *computational power to programmers using reinvented and restricted—but still recognizable—concepts of sequential code, integer arithmetic, pointers, and user-defined classes and objects. Within the space of indefinitely scalable designs, consequently, we prioritize programmability and software engineering concerns well ahead of either theoretical parsimony or maximally efficient or flexible tile hardware design.*

It's easy to read indefinite as "infinite" here. So what would such machine look like? Well, they've built the Movable Feast Machince as an implementation of their ideas:

> *We propose the indefinitely scalable Movable Feast Machine, which defers many architectural decisions to an execution model that associates processing, memory, and communications functions with movable bit patterns rather than fixed locations. We illustrate basic and novel computational elements such as self-healing wire, simple cell membranes for modularity, and robust stochastic sorting by movable self-replicating programs.*

> *The Movable Feast Machine (MFM) is a computer architecture and computational model under development by the Robust Physical Computation Group based at the University of New Mexico.The MFM is distinguished by its emphases on indefinite scalability and multilevel robustness. Compared to a traditional computer architecture that permanently assigns fixed functions -- CPU, cache, memory, bus -- to different bulk regions of a machine, the MFM is an open-ended*

> *spatial tiling of relatively small and locally-connected hardware elements, and it associates processing, memory, and communications functions with movable bit patterns rather than fixed locations.*
>
> *The Movable Feast Machine consists of a 2D grid in which each tile contains a processor with a fixed amount of volatile and non-volatile memory. Each processor can communicate with its nearest neighbor processors via point-to-point links. The computation model for the proposed machine consists of a set of "event windows" that involve a group of tiles communicating with each other to perform the computation. Note that many non-overlapping event windows can exist concurrently. One of the critiques for this proposed architecture is that the hardware costs will be too high for cost-effective computation.*

Sounds a bit like memristors meet the ambient cloud, which is why I find this idea so intriguing. It combines biological computing models, ideas from chemistry, and robust computing design.

Their presententation at HOTOS XIII has some interesting bullet points:

- Computation must be born again.
- Computation's Original Sin: Hardware Shall Be Reliable; Software Shall Be Efficient. Efficiency and robustness are mortal enemies.
- The solution: Indefinite scalability
- Sacrificing:
  - Fixed-width addresses, unique node names.

  - Logarithmic global communication cost
  - Clock and phase synchronization
- Embracing:
  - Opportunistic reproduction for ||ism & robustness
  - Movability for configuration, manifest destiny, ...
  - Multilevel robustness: Up to the end-user

How all this can be programmed to solve problem in real-life, I don't know. But I really like the notion of starting to see the all of computing as a single programmable fabric instead of islands of chunky servers. It's time to move the compute quanta down instead of up. Where this will lead, nobody knows. And they close their paper in that spirit:

> *And finally, to challenges on expressiveness—arguing the MFM will be too painful to program because our atom or event window is too small, or bonds too few, or lengths too short, or system dimensionality too low, we say: You might well be right. Let's find out.*

Related Articles

- Slide Deck for HotOS Presentation
- The Three Ages Of Google - Batch, Warehouse, Instant
- YouTube: A NAND gate in the Movable Feast Machine
- Move Feast Machine Home Page
- YouTube: Programming the Movable Feast Machine with λ-Codons
- SFB/Individual project pages
- HOTOS XIII Conference Report

http://highscalability.com/blog/2011/9/28/pursue-robust-indefinite-scalability-with-the-movable-feast.html

- Pursue Robust Indefinite Scalability With The Movable Feast Machine by Dave Ackley

---