

[SORS: Reverse-Engineering the Brain from a Computer Architecture Perspective](#)

Objectives



[Click here to see the presentation](#)

[Click here to see the updated version of the presentation.](#)

Bio: James E. Smith is Professor Emeritus in the Department of Electrical and Computer Engineering at the University of Wisconsin-Madison. He attended the University of Illinois, receiving his PhD in 1976. He then

joined the faculty of the University of Wisconsin-Madison, teaching and conducting research ? first in fault-tolerant computing, then in computer architecture.

In 1979 he took a leave of absence to work for the Control Data Corporation, participating in the design of the CYBER 180/990. While at Control Data and after returning to the University of Wisconsin in 1981, he studied several aspects of high performance pipelined processors. This work included the development of dynamic history-based branch predictors, instruction issuing methods, and techniques for providing precise interrupts that are widely used today. From 1984 to 1989, he was principal architect and a logic designer for the ACA ZS-1, a scientific computer employing a dynamically scheduled, superscalar processor architecture. In 1989, Dr. Smith joined Cray Research and headed a small research team that participated in the development and analysis of advanced supercomputer architectures. This work focused on advanced vector processor implementations, high bandwidth memory systems, and interconnection networks.

In 1994, he re-joined the Department of ECE at the University of Wisconsin. His research interests were directed at new paradigms for exploiting instruction level parallelism. The virtual machine abstraction was used as a technique for providing high performance through co-design and tight coupling of hardware and software. In 2007 he retired from Wisconsin, and then conducted research in industry for four years, first at Google, then at Intel. He received the 1999 ACM/IEEE Eckert-Mauchly Award for contributions to computer architecture.

For the past six years, he has been studying neuron-based computing paradigms at home along the Clark Fork near Missoula, Montana.

Abstract:

“The most powerful computing machine of all is the human brain. Is it possible to design and implement an architecture that mimics the way the human brain works?” -- Workshop report: Mary J. Irwin and John P. Shen, “Revitalizing computer architecture research,” Computing Research Association (2005).

Reverse-engineering the brain, so to speak, was identified as a grand challenge for computer architecture researchers over a decade ago, and there is no question that it is of immense consequence. Yet, computer architecture researchers have done remarkably little (virtually nothing) to address this challenge. One very significant reason is that experimental neuroscience has a long history and the neurobiological landscape is vast. Consequently, for someone outside the field, the mass of literature and diversity of theories is daunting, replete with controversy and contradiction. The learning curve is very steep and may appear insurmountable. It is difficult to even know where to start. The goal of this tutorial is to provide a starting point and significantly flatten the learning curve for researchers wishing to pursue the computer architecture grand challenge.

From the computer architecture perspective, the brain’s neocortex is a complex computing system that must be divided into a hierarchy of abstractions if it is ever to be understood. The way we use abstractions to model conventional computer systems serves as a guide for this “reverse-abstracting” process. At the bottom of the architecture stack, the first abstraction is from neuron biology to the lowest-level computing model. This is analogous to the abstraction from CMOS to binary logic. This first layer of abstraction is the foundation upon which everything else rests. Once away from the neurobiological domain, and in the computing model domain, higher levels of abstraction may be constructed in a well-defined manner.

In the past 20 years, theoretical neuroscientists have made significant progress in studying and architecting the lower levels of abstraction. This includes the initial biology-to-model abstraction, as well as the first abstraction layer: from single neurons to assemblies of neurons. This tutorial will bring attendees up-to-date

with neuroscientific progress toward “reverse-engineering the brain”, as *interpreted by a computer architect*. After attending the tutorial, attendees will: 1) better understand the nature of the problem, 2) view it as a computer architecture research problem, 3) have a firm toe-hold for initiating study of the problem, and, it is hoped, 4) participate in a serious effort to tackle the grand challenge!

Day 1-11/12/2017, 11.00h : biological background, time as a resource, neuron modeling

Day 2- 12/12/2017, 11.00h: temporal neural networks, neural architectures and case studies, simulators





Space-Time Computing Network



A Spacetime Computing System is a feedforward composition of functions
 $F_i: \langle x_1, x_2, \dots, x_n \rangle \rightarrow z; \quad x_i, z \in \{0, 1, \dots, \infty\}$

All F_i satisfy:

computability: F_i implements a computable total function.

causality: For all $x_i \geq z$, $F_i(x_1, \dots, x_n, x_i) = F_i(x_1, \dots, x_n)$, and if $z = \infty$, then $z \geq x_{i+1}$.

invariance: $F_i(x_1 + 1, \dots, x_n + 1) = F_i(x_1, x_n) + 1$

December 2017

copyright J.E. Smith

541

Barcelona Supercomputing Center - Centro Nacional de Supercomputación

Source URL (retrieved on 22 Sep 2024 - 01:07): <https://www.bsc.es/es/research-and-development/research-seminars/sors-reverse-engineering-the-brain-computer-architecture-perspective>