Argonne NATIONAL LABORATORY

From Exascale Software to Internet of Things

We are thinking to small…
&
Can IOT learn from Exascale?

Pete Beckman

Argonne National Laboratory

Northwestern University

U.S. DEPARTMENT OF ENERGY

PROJECT ARRANGEMENT
UNDER THE IMPLEMENTING ARRANGEMENT
BETWEEN
THE MINISTRY OF EDUCATION, CULTURE, SPORTS, SCIENCE AND TECHNOLOGY OF JAPAN
AND
THE DEPARTMENT OF ENERGY OF THE UNITED STATES OF AMERICA
CONCERNING COOPERATION IN RESEARCH AND DEVELOPMENT IN ENERGY AND RELATED FIELDS

CONCERNING COMPUTER SCIENCE AND SOFTWARE RELATED TO CURRENT AND FUTURE HIGH PERFORMANCE COMPUTING FOR OPEN SCIENTIFIC RESEARCH

* This is excerpt from the project arrangement

Technical Areas of Cooperation

- Kernel System Programming Interface
- Low-level Communication Layer
- Task and Thread Management to Support Massive Concurrency
- Power Management and Optimization
- Data Staging and Input/Output (I/O) Bottlenecks
- File System and I/O Management
- Improving System and Application Resilience to Chip Failures and other Faults
- Mini-Applications for Exascale Component-Based Performance Modelling

Blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah *Exascale Challenges* blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

- Are we tired yet?

- Where are the ideas changing abstractions?
- Where are we changing the model?

Argonne NATIONAL LABORATORY

An Exascale Operating System and Runtime Research Project

ANL, LLNL, PNNL, UTK, Uoregon, Uchicago, UIUC

+ Industry advice…

U.S. DEPARTMENT OF ENERGY

COMMUNICATION COMPLEXITY OF PRAMs

Alok AGGARWAL, Ashok K. CHANDRA and Marc SNIR

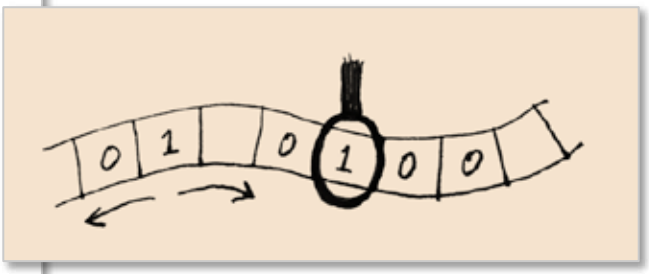
IBM Research Division, T.J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598, USA

Abstract. We propose a model, LPRAM, for parallel random access machines with local memory that captures both the communication and computational requirements in parallel computation. For this model, we present several interesting results, including the following:

Two $n \times n$ matrices can be multiplied in $O(n^3/p)$ computation time and $O(n^2/p^{2/3})$ communication steps using p processors (for $p = O(n^3/\log^{3/2} n)$). Furthermore, these bounds are optimal for arithmetic on semirings (using $+$, $\times$ only). It is shown that any algorithm that uses comparisons only and that sorts n words requires $\Omega(n \log n/(p \log$ We also provide an algorithm that sorts n words an $\Theta(n \log n/(p \log(n/p)))$ communication steps. These FFT graph.

It is shown that computing any binary tree τ with n $\sqrt{h})$ communication steps, and can always be compu present a simple linear-time algorithm that generat $2D_{opt}(\tau)$ steps, where $D_{opt}(\tau)$ represents the minimu

It is also shown that various problems that are exp



A More Practical PRAM Model

Phillip B. Gibbons
Computer Science Division
University of California
Berkeley, CA 94720

Abstract

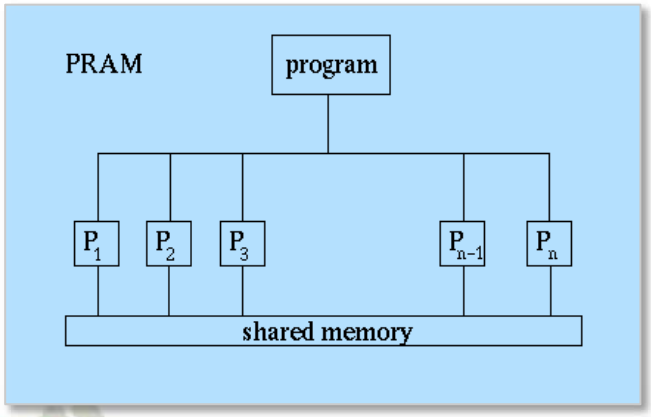
This paper introduces the Asynchronous PRAM model of computation, a variant of the PRAM in which the processors run asynchronously and there is an explicit charge for synchronization. A family of Asynchronous PRAM's are defined, varying in the types of synchronization steps permitted and the costs for accessing the shared memory. Algorithms, lower bounds, and simulation results are presented for an interesting member of the family.

There are several difficulties that arise in mapping PRAM algorithms onto existing shared memory MIMD machines, such as the Sequent Balance, the BBN Butterfly, the NYU Ultracomputer, and the IBM RP3. First, realistic MIMD machines have more limited communication capabilities than the PRAM. The PRAM assumes that each processor can access any memory location in one step. Realistic machines are more limited in at least three respects:

1 Intro

The PRAM
tion of p sequential processors, each with its own private local memory, communicating with one another through

banks by an interconnection network in which the shortest path between a processor and most loca-



Abstractions Matter

For Extreme-Scale Systems:

- Nodes are no longer simple
  - The Node OS is no longer suitable for PRAM

- Systems are no longer simple
  - A simple set of “nodes” is insufficient

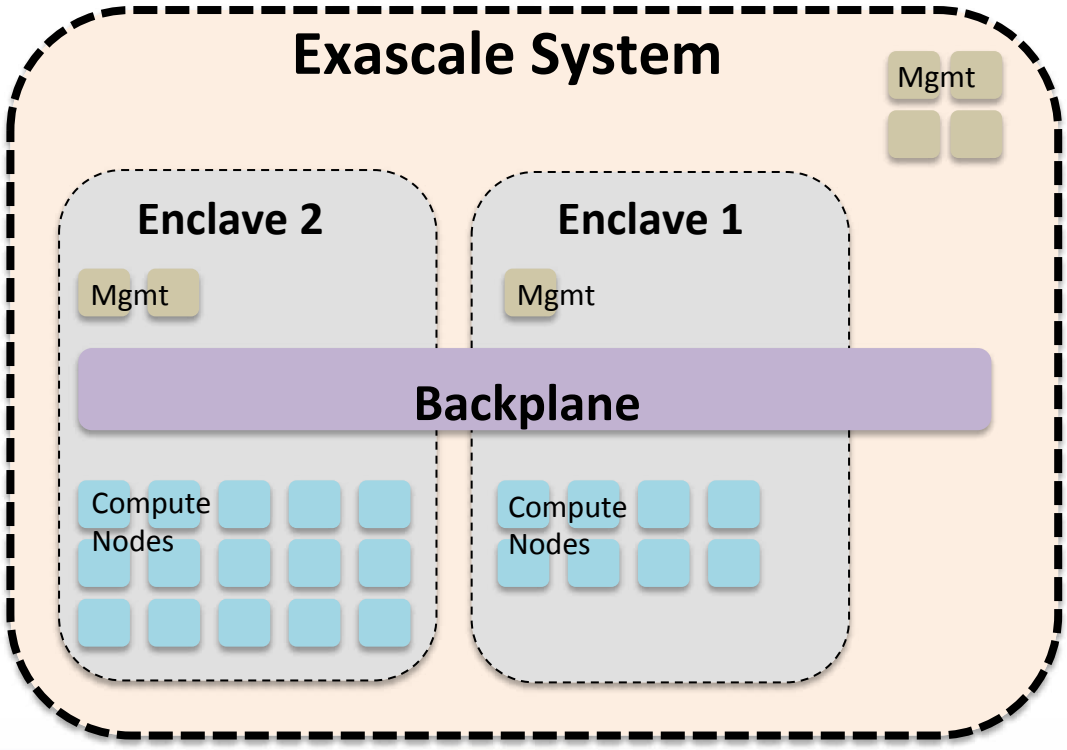
- Enclave
  - (recursive)
  - tree-based hierarchy and recursive decomposition
  - At each level in the hierarchy, four key aspects change: granularity of control, communication frequency, goals, and data resolution.

New Definition for "Program"

- <Application Exec>
- <Enclave Mgmt Exec(s)>
- <Backplane Events>

Enables a

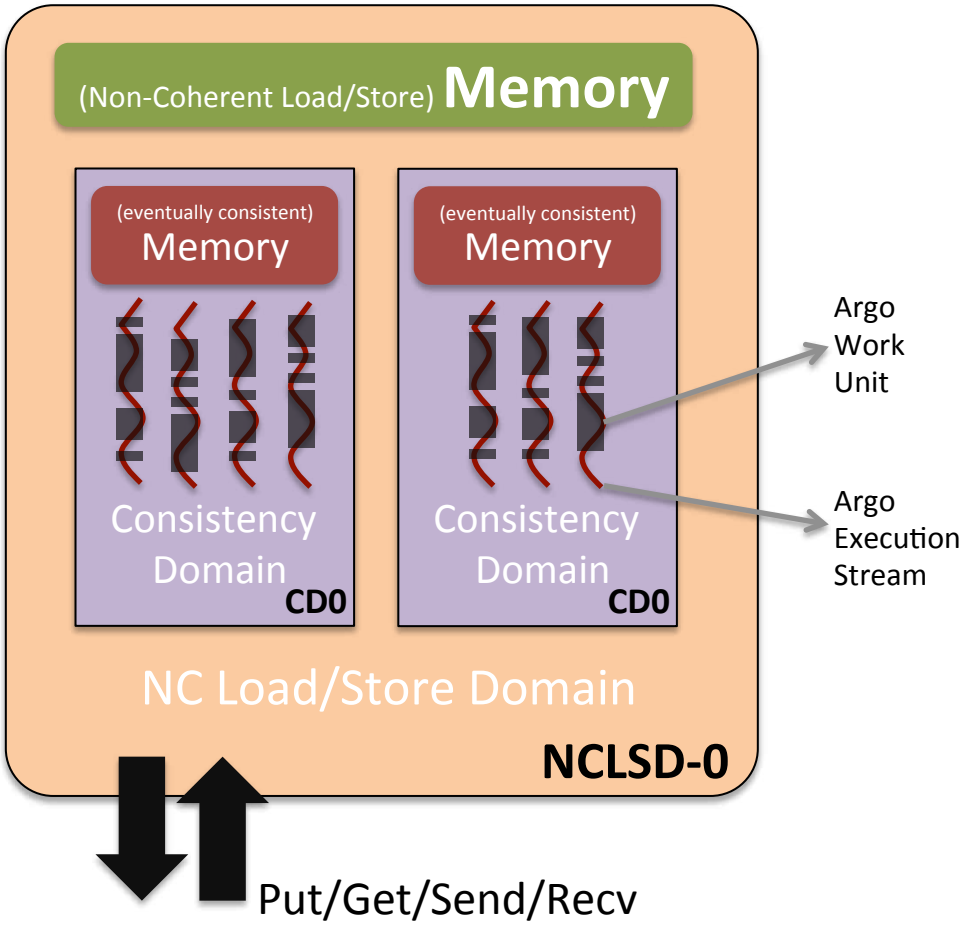
Global OS/Runtime



Benefits

- Embedded feedback and response mechanisms
  - *Self-aware, Goal-based active run-time systems*
- Meta-handle for enclaves
  - Can write meta-programs for enclave
    - (manage parallelism, task-manager, etc)
  - Allows application-specific fault managers, streaming I/O handlers, many-task UQ engines, and event-based coordination of coupled components
- Hierarchical, coordinated, global system
  - manage power budgets, respond to faults
  - support enclave components for machine learning

(Simplified) Argobots Abstract Machine

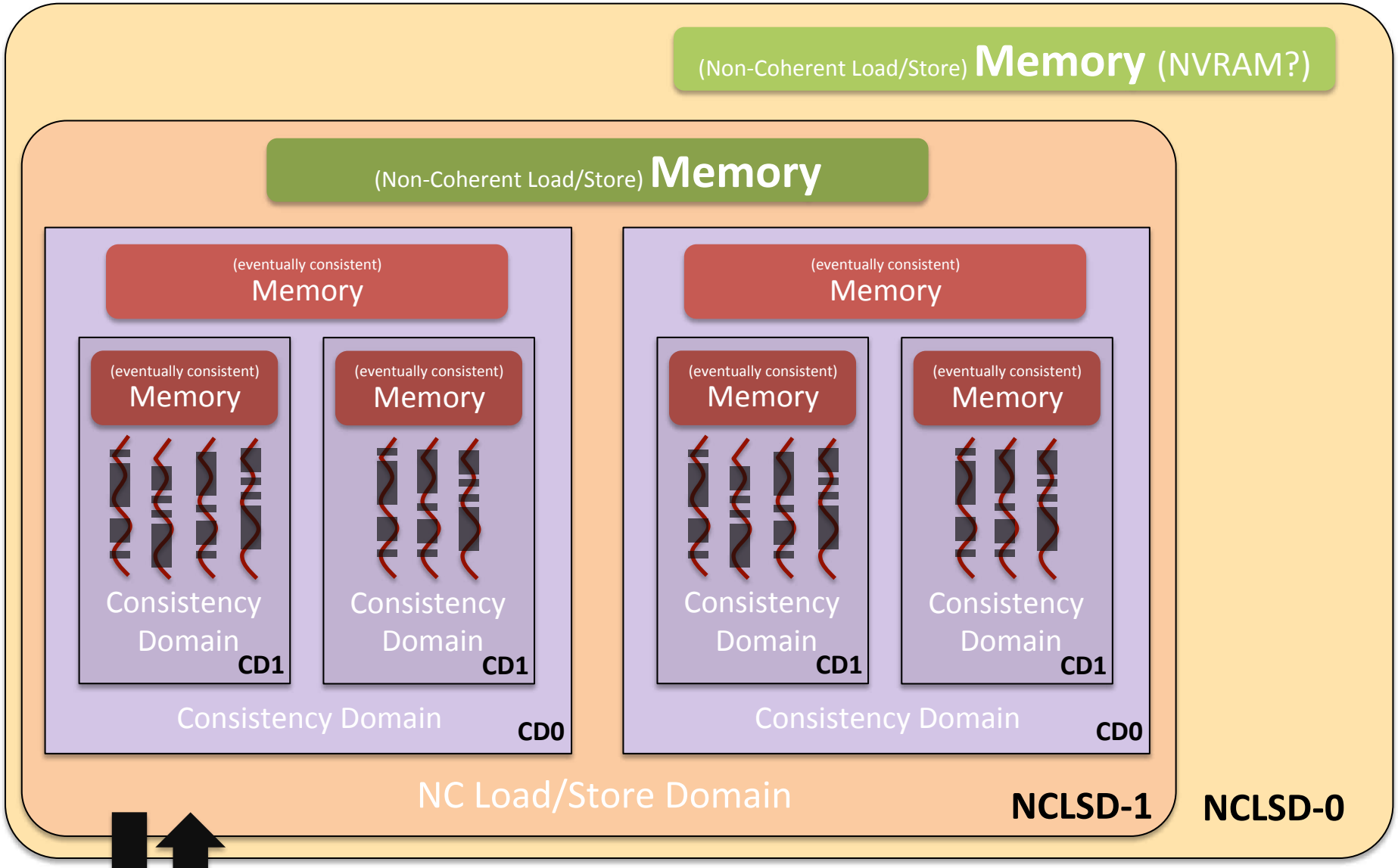


Key Concepts

- Separation of abstraction and mapping to implementation
- Massive parallelism
  - ***Thread****s can yield*
  - ***Tasklets*** execute to completion
  - ***Exec. Streams*** guarantee progress
- Clearly defined memory semantics
  - Eventual Consistency
  - Common virtual addressing and software-managed consistency
  - Support explicit memory placement and movement
- Put/Get/Send/Recv requires library call in OSR
- Exploring fault model and atomics

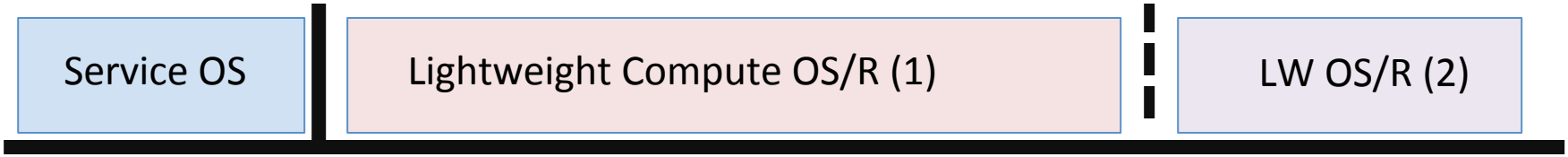
A foolish consistency is the hobgoblin of little minds […]
Ralph Waldo Emerson

(Generalized) Argobots Abstract Machine



"Programmer" is high-level library, language, or compiler writer, not domain scientist

Node Operating System and Runtime



- Small-core-count ServiceOS for overall node management
- Different compute instances for running application code
  - POSIX support provided by a Linux-based kernel
  - Argo runtime: dedicated, tightly integrated kernel or Linux containers
  - Framework will be generic so other kernels and runtimes can be used
- Hardware resources ***partitioned*** between OS instances
  - individual instances limited to sizes that scale well/do not cross coherence domains

We are thinking to small…

Internet of Things? Yawn?

- 1993: First Internet webcam to check on coffee pot

- 2014: Internet connected lightbulbs, thermostats, etc

- Wow…. Impressive…. ???

USA TODAY
A GANNETT COMPANY

Search

SUBSCRIBE NOW
3 MONTHS FOR
THE PRICE OF 1

NEWS | SPORTS | LIFE | MONEY | TECH | TRAVEL | OPINION | WEATHER | MORE

Home Depot expands stock of smart home gadgets

Wendy Koch, USA TODAY 5:58 p.m. EDT July 7, 2014

(Photo: Mikki K Harris)

f 16 CONNECT | 98 TWEET | in 19 LINKEDIN | 2 COMMENT | EMAIL | MORE

Boosting your home's IQ got easier Monday as The Home Depot began selling a collection of nearly 60 gadgets that can be controlled by mobile devices, including light bulbs, lawn sprinklers and water heaters.

Now that mobile devices can remotely operate appliances and other items, the number of smart-home products is exploding. Two years ago, The Home Depot sold 100 of them but now offers 600, said Jeff Epstein, the retailer's vice president for home automation merchandising.

Nearly all of its 2,000 U.S stores and its website will now be selling dozens of them with software developed by Wink, a company spun off from New York start-up Quirky.

gadgets, home auomation, internet of things

Samsung, Intel, Dell team on Internet of Things connectivity standards

Some top hardware companies have established a new Internet of Things consortium to create standards so that billions of devices can connect to each other.

Intel, Samsung and Dell are among the founding members of Open Interconnect Consortium (OIC), which later this year will deliver the first of many specifications for hassle-free data flow between devices, regardless of the OS, device type or wireless communication technology.

The OIC companies will contribute open-source code so developers can write common software stacks for communications and notifications across handsets, remote controls, wearables, appliances and other sensor devices.

The consortium will first establish standards around connectivity, discovery and authentication of devices, and data-gathering instruments in “smart homes,” consumer electronics and enterprises, said Gary Martz, product line manager at Intel.

What Are We Missing?

Internet of Computing Things!

- Hypothesis:
  - When we move real ***computation*** into the sensing platforms, we will see a transformation.

- What will be the OS, system software, and programming model for Internet of Computing Things?
- Are there lessons from Exascale?

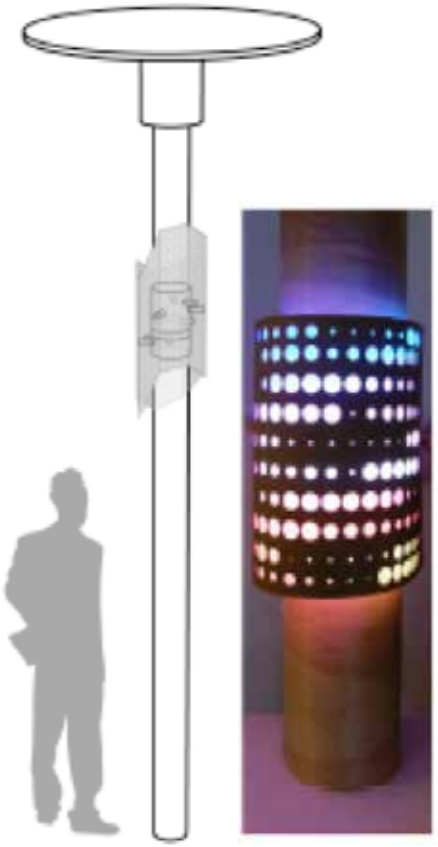
1.7GHz 4-Core
2GB RAM
Xubuntu or Android
Price : US$65.00

ODROID

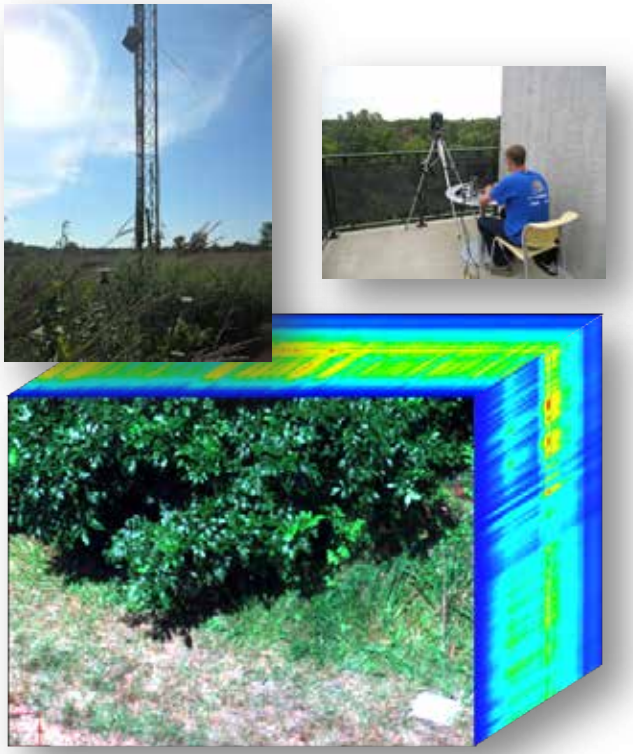
1.6GHz 17-Core
16GB RAM
Difficult non-Linux OS
Price : (a laptop)

BG/Q

Weather / Flux

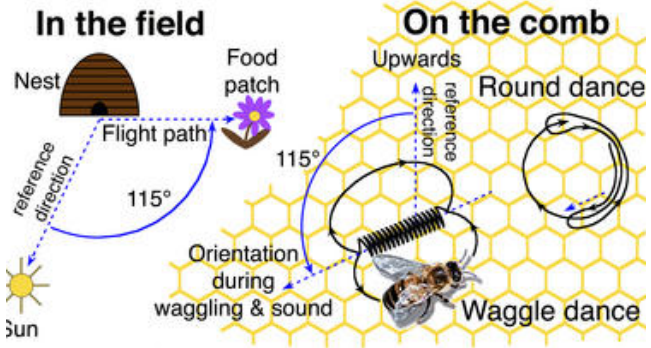


Chicago Urban Data



Hyperspectral Imaging (e.g. 20GB/day)

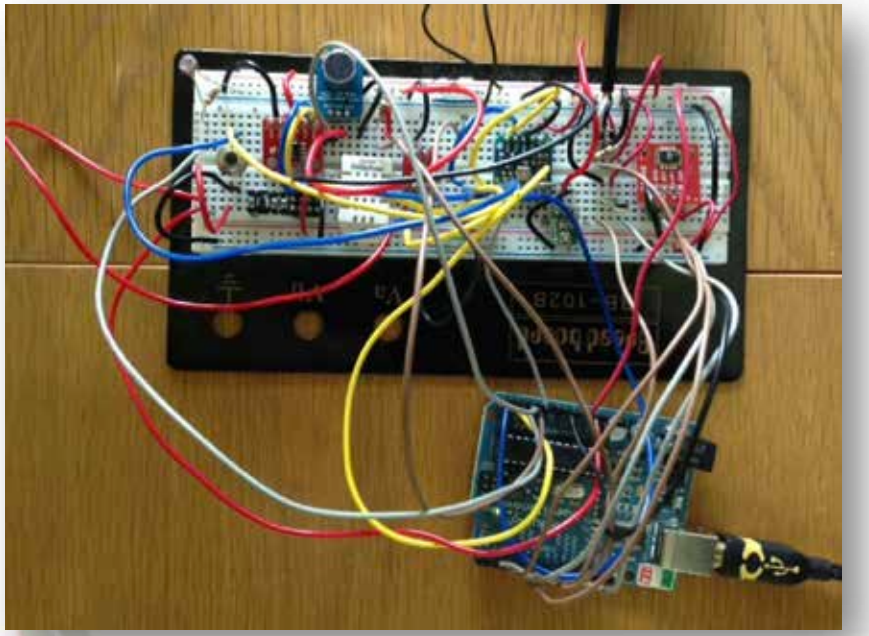
Waggle: A System Software Framework for Computing in Sensor Platforms

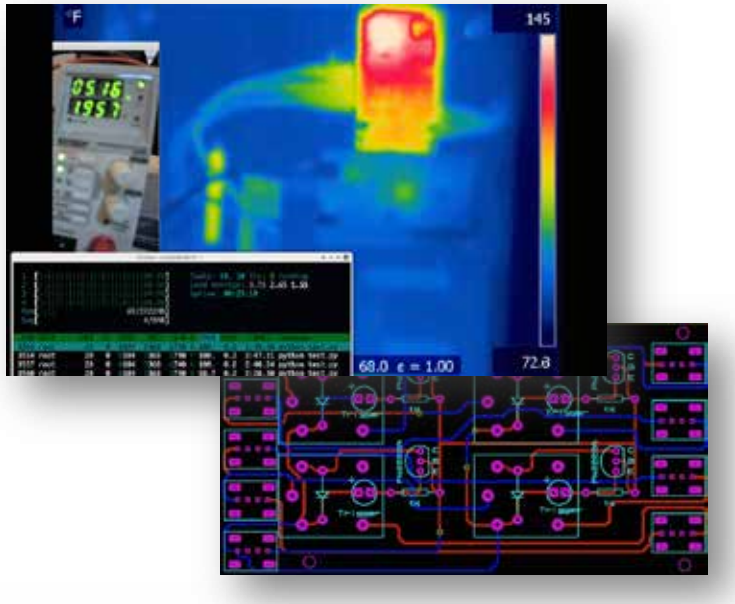


- Why Now?
  - Possible to build "Attentive Systems"
  - Explosion of sensors, actuators, remote sensing platforms
  - Plenty of low-power computing available locally
  - New machine learning, data analysis, and classification algorithms for large multi-dimensional data
- What can we leverage from HPC?
  - Fault tolerance? (who learns from whom?)
  - Power-aware computing? (who learns from whom?)
  - Scalability: Collectives, Hierarchy, Management
  - Parallel programming, Data movement reducing algs.

Design:

- Planning for “Deep Space” deployment
  - fault tolerance, power, remote management
- Hierarchy, tree reductions, collectives
- Argo-like design: “Service OS” partitioned computing
- Cloud-backed data for streaming to other places
- Prototype today in lab, will be test deployed in Sept





NASA ISEE-3, 36 years old

Pondering Security and Privacy….

(we don't generally like to spend too much time on this in HPC)

Level 0	Level 1	Level 2	Level 3
			
Hardware Limited Resolution	***Read-only*** Firmware Limited Resolution	Remote Key / Crypto ***Modifiable*** Firmware Limited Resolution	***Modifiable*** Software Privacy from security

Future

ARGO:

- Build OS/R for extreme-scale systems
- Push envelope on abstractions for OS/R, "program", and dynamic adaptable global OS/R
- Look at what we can learn from embedded computing

Waggle:

- Deploy Waggle systems in Chicago, at Argonne, at UIC, etc.
- Release Open Framework for "Internet of Computing Things"
- Explore using framework for next generation environmental sensing, urban data, weather, metagenomics, UAV
- Apply scalability ideas from HPC, learn about fault tolerance

Questions?