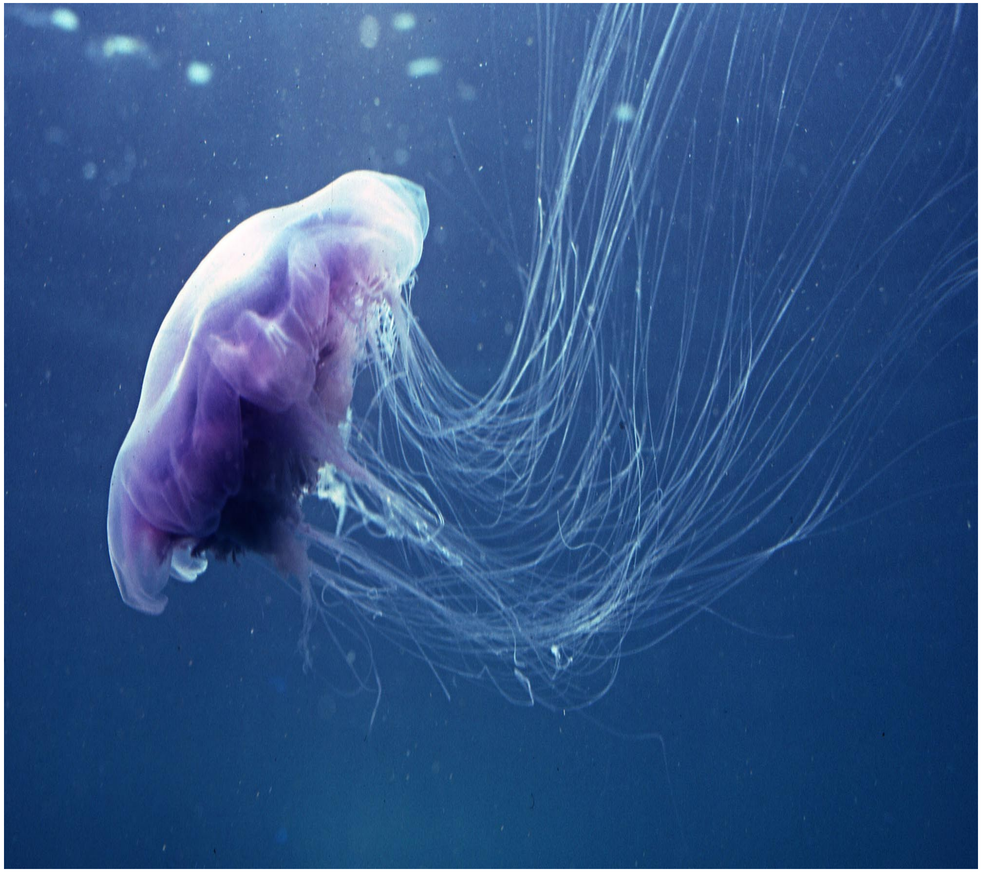
Evolution of Multicellular Computing: Parallels with Multicellular Life

Seminar, December 21, 2009

University of Birmingham, UK

Steve Burbeck

evolutionofcomputing.org



What would you do with a Multicellular Computer?

Suppose that in 2012, 11 lorries (trucks) are set to arrive at your data center each carrying a shipping container

10 of the lorries carry a data-center in a box (from IBM, Sun, Dell, HP, or other vendors):

- Each box contains 2,000 48-core servers. The total for the shipment is nearly one million servers.
- Each container requires 3 connections:
  1) a fiber-optic data link,
  2) 400KW of power, and
  3) 60 tons of chilled water cooling.

The 11th box contains a 600 ton water chiller that can cool the 10 computing containers

Together the eleven containers use over 4 megawatts.

The combined information processing capability is equivalent to that of a small jellyfish.

What do you do? What does it do? How do you “program” it? How do you keep it from quickly becoming taken over by a botnet?

Why the Multicellular Metaphor?

- The “Cloud Computing” metaphor isn't about much of anything...its basically a marketing term that doesn't offer useful architectural insights.

- Parallel computing is about parallel algorithms and maximum FLOPS. It does not offer much insight about what to do with a million independent servers, or how to do it.

- Swarm computing gets much closer (and we will talk a bit about that). But it lacks important organizing principles used by multicellular organisms.

- The Multicellular metaphor is the subject of the rest of this talk.

Existing Large Multicellular Computing Systems

Google

- At least 9 specialized types of server including crawers, ad servers, indexing, spelling, documents, http, search, formatting, proxy/cache
- A total of at least 1.5 million servers – 1160 per shipping container, 45 containers per data-center warehouse, and at least 35 data centers world-wide

Facebook

- 30,000 servers of at least 4 types

Ebay

- At least 5 functional types of servers: Database, LDAP, Web Servers, Application Servers, Networking Switches and Routers
- Estimated 15K-20K servers

Others

- Large public database services (e.g., EMBL/EBI Hinxton GenBank)
- Cloud systems – Google's Cloud, Amazon's cloud (EC2), Microsoft's cloud, Salesforce.com
- Social networks (e.g., MySpace, Facebook, Twitter)
- Massively Multiplayer Online Role Playing Games (e.g., World of Warcraft, EverQuest, Second Life),
- Instant messaging chat and VOIP systems (e.g., Skype)
- Wikipedia

The Analogy Between Life and Computing

- The forces that have shaped the evolution of computing are similar to those in the evolution of life:
  - As complexity grows, multiple levels of organization or abstraction (meta-levels) emerge
  - Subsystems become encapsulated to reduce unwanted or dangerous interactions
  - Species co-evolve in response to each other to improve predation, symbiosis and other co-dependent behavior.
- Evolution in computing architecture occurs both despite and because of our attempts at "design." The history of computing provides insight into the circumstances of the emergence of computational meta-levels.
- As computing continues to become more complex and powerful, we can and should apply lessons learned from 3.5 Billion years of trial and error biological experimentation.

The Central Issues

Emergence of meta-levels

- Emergence is known variously as self-organization (Kauffman), metasystem transition (Turchin), autocatalysis (Prigogine), increasing returns and path dependence in economics (Brian Arthur), and others.
- Living and digital systems cannot help but undergo the stepwise emergence of multiple meta-levels

Multicellularity is a major organizational transition – what are its prerequisites?

- Single cells and single computers are limited in scope, scale and complexity.
- Multicellularity offers solutions to those problems.
- Yet multicellular systems face new issues such as cooperation, protection from predators, and maintenance of “self”

Strategies that support multicellular systems

- The most basic are: Specialization, Orchestration (via Messaging & Stigmergy), and Self Protection (Apoptosis together with Stigmergy)
- These strategies predate or are coincident with the multicellular transition
- And they are applicable to multicellular computing in the Internet

Ubiquitous Information Processing

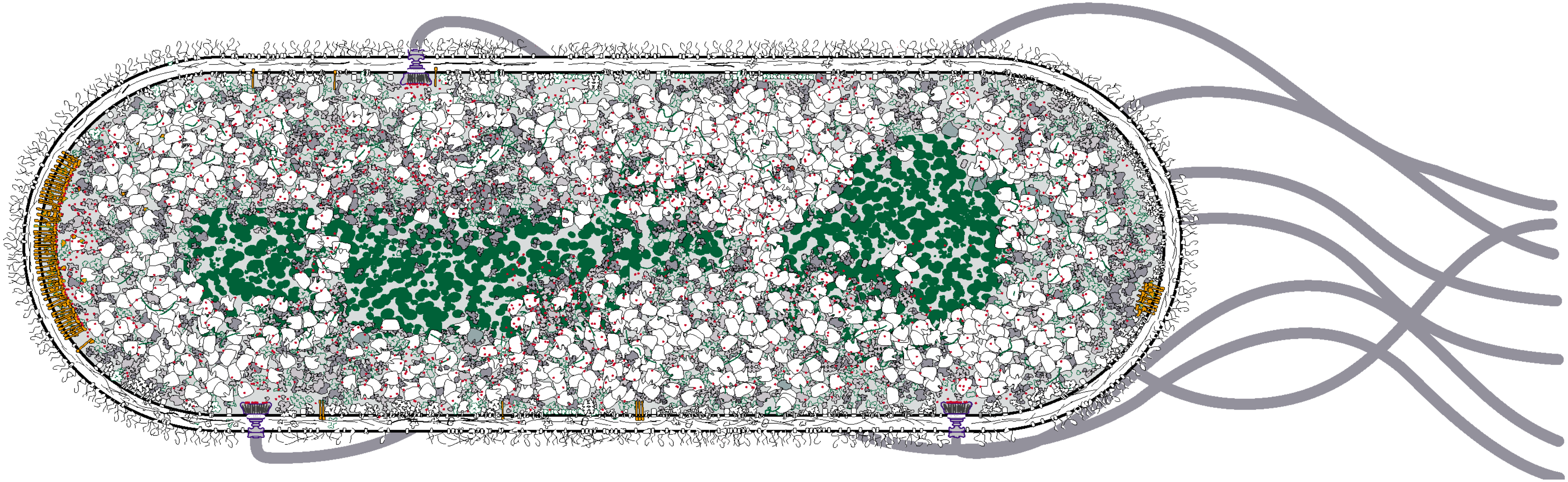
All living organisms sense, process, and act on information in their environment.

- Many ***single-cell*** organisms sense light, vibration, temperature, and their chemical environment. They respond to changing sensory input by modifying their internal behavior and/or moving to more favorable locations.
- Many organisms (including single cell bacteria) “signal” others in a way that promotes cooperative behavior. It's called ***“quorum sensing”*** in bacteria.
- Even plants cooperate. Some plants sense attack by insects, increase their defensive chemistry, and emit pheromones that signal attack to other nearby plants.
- We don't necessarily know what information a given organism senses, what the result of the processing determines, what time-scale it operates on, or what “purpose” it serves.

Digital information processing surrounds us but is largely invisible, even to computing professionals.

- Much of what modern societies and economies do is mediated by digital information processing
- If you are not a computing professional, you may not perceive or even know abstractly the digital processing taking place all around us.
- We swim in a sea of electromagnetic radiation that today is mostly produced by and received by digital devices of many sorts.
- We seldom are far away from wires, fiber optic cables and wireless signals.

Example of Biological Information Processing



http://www.pdn.cam.ac.uk/groups/comp-cell/Big_bug.html

Escherecia coli (E. coli) is one of the simplest and best studied bacteria. Biologists know a great deal about how it senses, processes and responds to information in its environment

Sensors (yellow) provide information about various attractant and repellent aspects of the environment

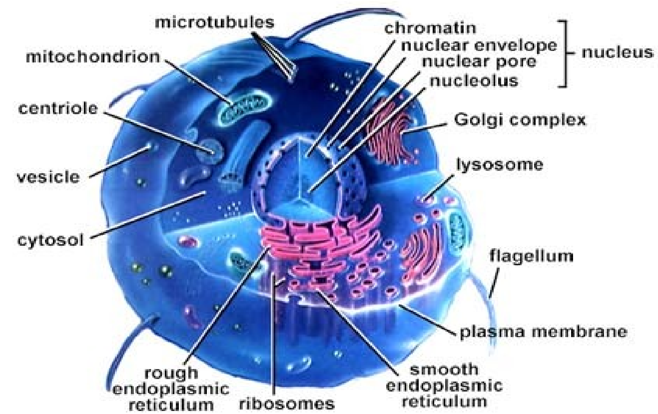
- Individual sensor proteins have a 10^2 dynamic range
- By cooperation between sensors, the patch as a whole provides a 10^4 or 10^5 range

Messenger proteins (CheYp, represented by small red dots) diffuse throughout the cell and are simultaneously dephosphorylated (i.e., inactivated) by complex feedback mechanisms.

Messengers bind to and regulate 6 - 8 flagella with bi-directional motors

- When turning counterclockwise, the flagella bunch together to propel the bacteria forward
- When turning clockwise, the flagella separate and cause the cell to tumble randomly
- When attractant dominates, forward runs tend to be relatively long, when repellents dominate tumbles are more frequent

Information Processing Comparison: Standalone Computer vs. Single Cell Organism



A few GB RAM (code and data)

Persistent data in multi-GB Disk

One or a few Multi GHz serial processors

Deterministic and fragile – needs frequent reboots, little tolerance for randomness, can be disabled by failure of single gates.

Poor ability for self organization, no ability to replicate, dependent on outside power, maintenance and support.

Notoriously susceptible to malware attack

Code size (DNA) – from 5MB (simple bacteria) to 75GB (Amoeba). Note: human has ~1GB

Persistent data encoded in protein structure and DNA

Thousands to millions of KHz “processors” (protein and RNA molecular machines) operating in parallel

Stochastic and robust – Cells tolerate, even exploit, randomness. Robustness far exceeds human-engineered redundancy

Self organizing, self supporting, self healing, and self reproducing

Has substantial defenses against infection – even bacteria have enzyme defenses against viral infection!

Evolution of Computing

Early computers were a small, simple, slow collection of discrete logic gates

- The unit of software was the machine instruction – no abstraction whatever and no tools other than "coding sheets"
- I/O consisted of little more than panel display lights or a teletype.

Computing professionals alive today witnessed the following evolution

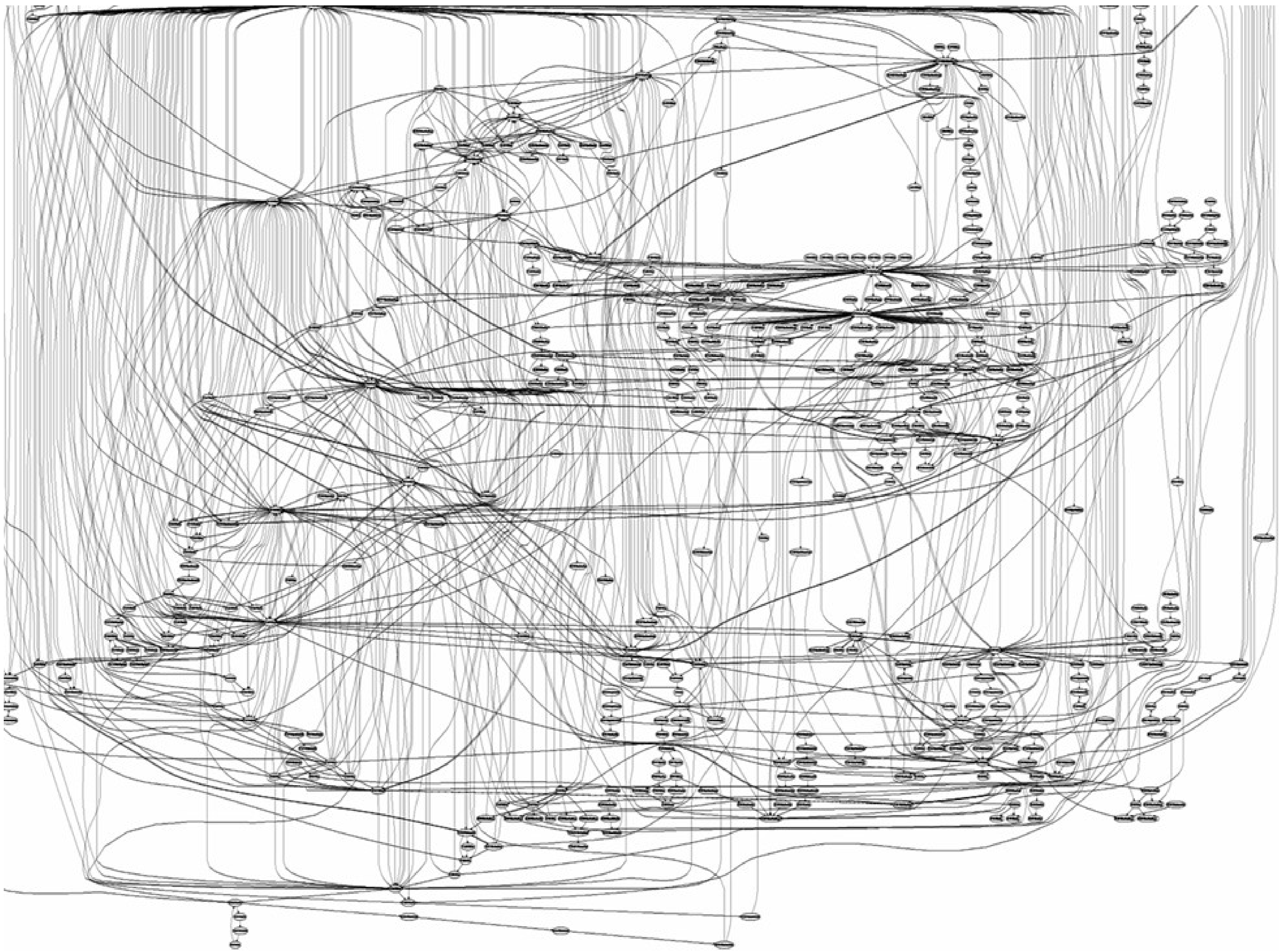
- Hardware evolved protected address spaces, virtual memory, microcoded instruction sets, "privileged instructions," pipelined processors, parallel and multi-core processors, various external storage media, various sorts of physical memory, etc.
- "Systems Software" evolved to provide Operating Systems, Virtual Machines, and I//O driver architectures
- Ad hoc file structures evolved into databases
- Software languages, assemblers, compilers and tools emerged and evolved.
- Programming genres proliferated – procedural, functional, object-oriented, ...
- The notion of what a computer is "for" evolved beyond arithmetic to include text processing, image processing, signal processing, real-time machine controllers, audio and video, file-sharing, … , now to social networking
- Business models evolved: leasing hardware, selling software, … , to selling "eyeballs" and Internet data plans.
- Interconnection models evolved from teletypes and paper tape, to RS-232, to the Internet built on fiber optics and wireless

Co-Evolution & Complexity

- Co-evolution in life makes strange bedfellows such as "Bee Orchids" that mimic insect pheromones to trick bees into pollinating the orchid
- Computing co-evolves via a human-driven spiral of technological, social, economic, cultural and conceptual novelty
  - The Wintel duopoly is a co-evolutionary relationship in computing
- Co-evolution drives increases in computing function and complexity
- Increasing complexity forces new architectures and the emergence of new levels of abstraction, cooperation, and competition

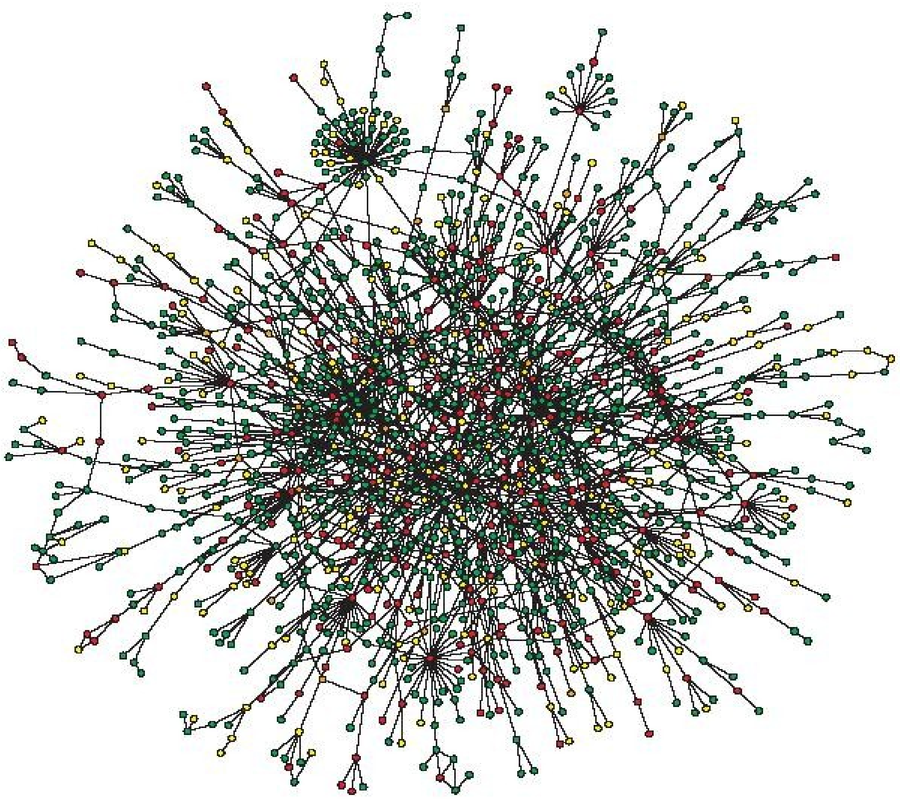
"Bee Orchid" with pseudocopulating bee

Software Complexity (Windows IIS)



IIS call graph for serving a single HTTP page containing one image

Complexity of Single Cell Protein-Protein Interactions (Yeast)



Yeast S. cerevisiae protein-protein interaction diagram. Barabasi & Oltvai, Nature Reviews Genetics 5, 101-113 (February 2004)

Each point represents a type of protein.

Each protein molecule has multiple states that affect its function

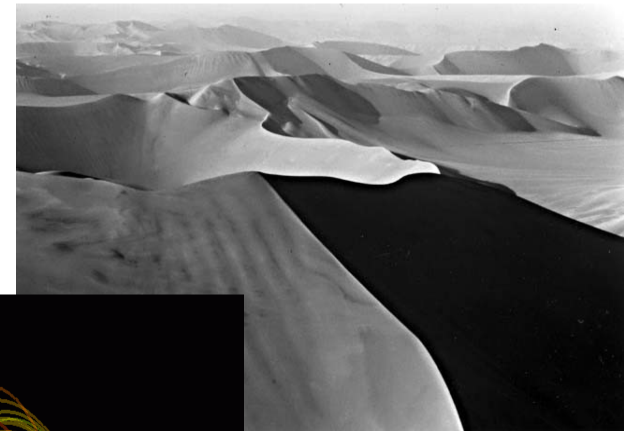
Lines that connect proteins signify that they bind to each other for one or more functional purposes such as structure, movement, signaling, memory, energy metabolism, or reproduction.

At least 2,358 known types of interaction occur among at least 1,548 types of protein in a yeast cell.

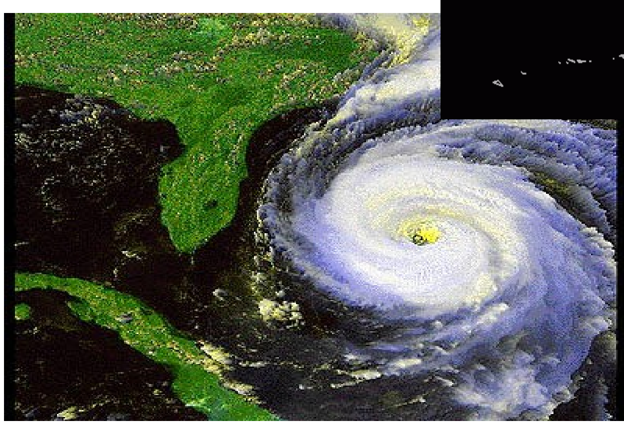
There may be only a few hundred copies of a given regulatory or signaling protein or millions of copies of a structural protein.

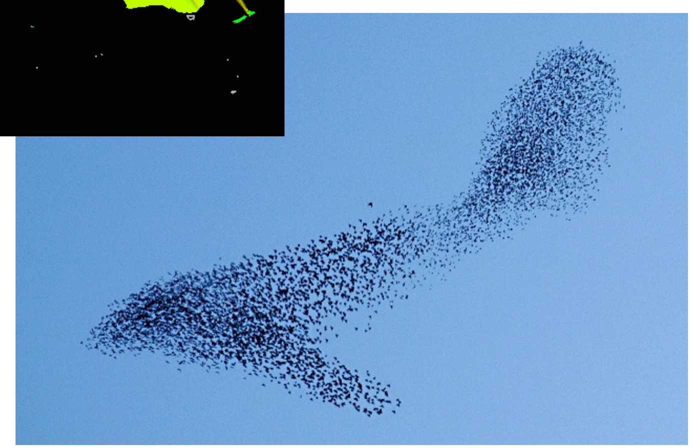
Proteins are only part of the cellular machinery – nucleotides, sugars, lipids, small inorganic molecules and small organic molecules play important functional and signaling roles as well.

Familiar Emergent Phenomena



The "long haul" Internet





Emergence in Biology and Computing

- Complex physics begets complex biochemistry begets the complex cells that collaborate to form complex organisms
- Complex digital hardware begets complex software which begets complex interactions between computers, especially in the Internet
- This “begetting” is emergent, i.e., self-organizing, driven by positive feedback between interacting sets of elements
    - Self-reinforcing sets become elements in a meta-level system
    - As each meta-level emerges, its behavior is qualitatively different from that of its constituent elements
    - Each new meta-level typically encapsulates, or hides, details (information) about the lower level phenomena
    - The organizational structure of each new meta-level creates new information (i.e. structural order) at the higher level
- The higher level behavior can often be thought of as a “virtual machine” interface that is “implemented” by the coordinated action of its lower level elements.

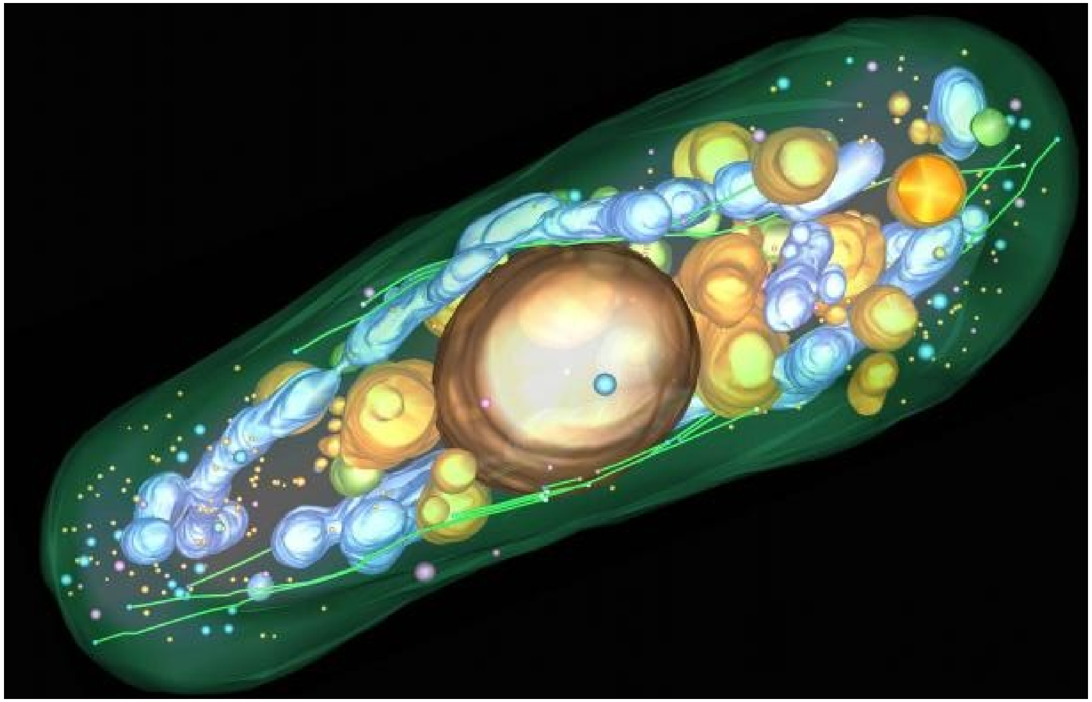
Causality Crosses Meta-Levels Imperfectly

- Causality crosses almost all meta-level boundaries, both upward and downward, but typically imperfectly and in ways difficult to predict.
- Upward causation is due to details of how lower level elements play a role in specific autocatalytic sets. Meta-level behavior tends to be robust, so lower level "causes" may only weakly affect the higher-level behavior. But in some cases, effects at higher levels can be dramatic:
  - Very low-level code can create buffer overruns in Windows that have Internet-wide consequences
  - Pharmaceutical drugs directly affect individual cells that indirectly affect the brain, hence psychological states such as depression.
- Downward effects are usually due to induced organization of lower level elements (i.e., added information) created by the upper level. Examples:
  - flocks, where visible "flockness" guides flight of individual birds. Real-time events act as signals
  - Hives, termite mounds, databases, … provide more persistent organization which may make it easier to understand "causes" and "effects" separately

Countering Unwanted Interactions with Encapsulation – The Designer Perspective

- Cells evolved many sorts of membrane encapsulated subunits long ago:
  - The cell membrane itself which prevents most external chemicals from entering or leaving the cell.
  - The nucleus, which encapsulates the DNA and basic gene expression mechanisms (comparable to OS ring 0)
  - Mitochondria which encapsulate energy transduction
  - Endoplasmic Reticulum that encapsulates much of protein production
  - And large molecular machines organize and orchestrate dozens of molecular parts so that they operate only when and where they are “supposed” to.
- Architects/designers in computing create similar encapsulation barriers “by design” to minimize unwanted interactions that all too often generate unwanted emergent behavior (bugs)
  - In hardware and operating systems, encapsulation barriers include multiple address spaces, privilege rings, “root” access, etc.
  - Barriers in software to unwanted interaction are created by mechanisms such as scoped variables, stack-based addressing, process boundaries, and virtual memory.

Encapsulation in Single-Cell Yeast



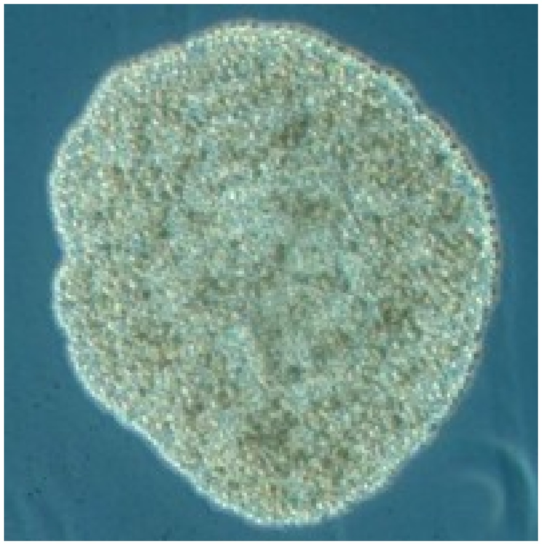
Electron tomogram of a complete yeast cell shows many internal membrane enclosures that prevent unwanted interactions.

Hoog JL, Schwartz C, Noon AT, O'toole ET, Mastronarde DN, McIntosh JR, Antony C. Organization of interphase microtubules in fission yeast analyzed by electron tomography. Dev Cell. 2007 Mar;12(3):349-61.

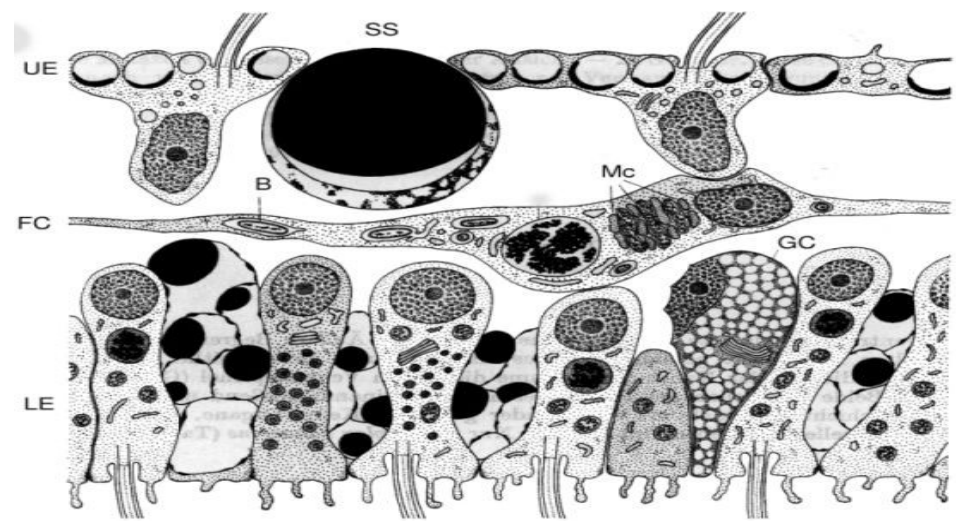
Multicellularity

- Biofilms (training wheels for true multicellularity) consist of many cooperating bacteria and viruses, often of many species, with separate and differing DNA. The cells temporarily form a multicellular organism by changing their behavior when they sense that they are in a sufficiently large group. This is called "quorum sensing." Because the cooperative behavior is not genetically unified, evolution of more complex behavior is problematic.
- Multicellular organisms, or Metazoans, have multiple cooperating cells that share the same DNA. As each organism grows, cells differentiate into specialized cell types that each use only part of the DNA. Differentiation itself is under DNA "program control" so the success or failure of the whole organism is determined both by the success of the various specializations and by the success of their orchestration.

Placozoa: the Simplest Metazoan

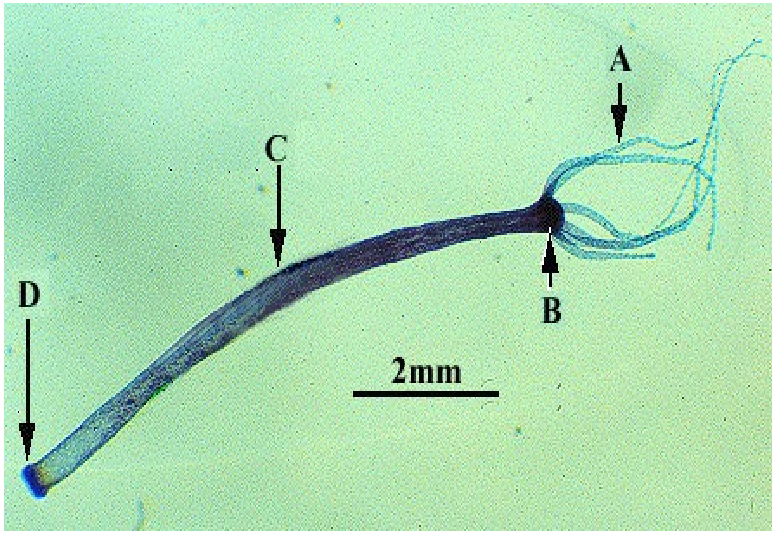


- Millimeter-scale discs that contain a few thousand cells of only four types. Dorsal cells with flagella, ventral “gland cells”, ventral flagellar cells, and central “fiber cells. Fiber cells also contain symbiotic bacteria.
- Moves in coordinated manner using the central layer of “fiber cells” that have both neuron-like and muscle-like function.
- Number of cells and cell-types ***comparable to eBay***

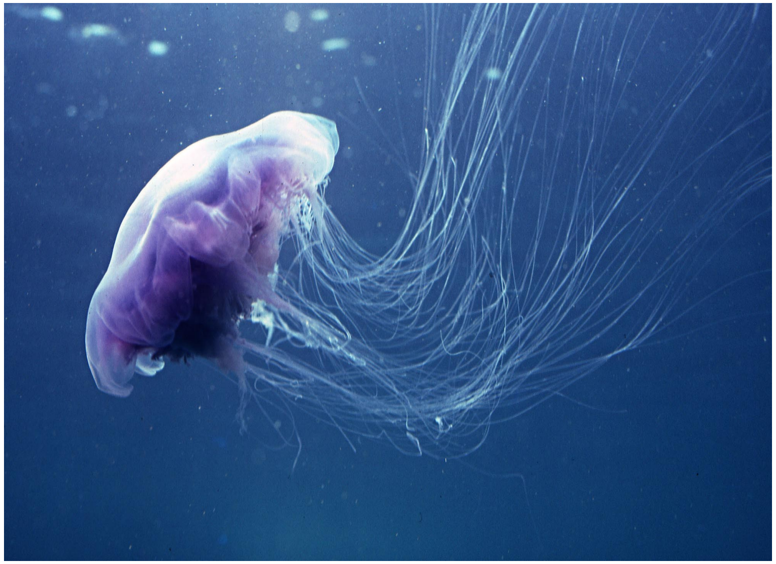


Cnideria – Considerably More Complex

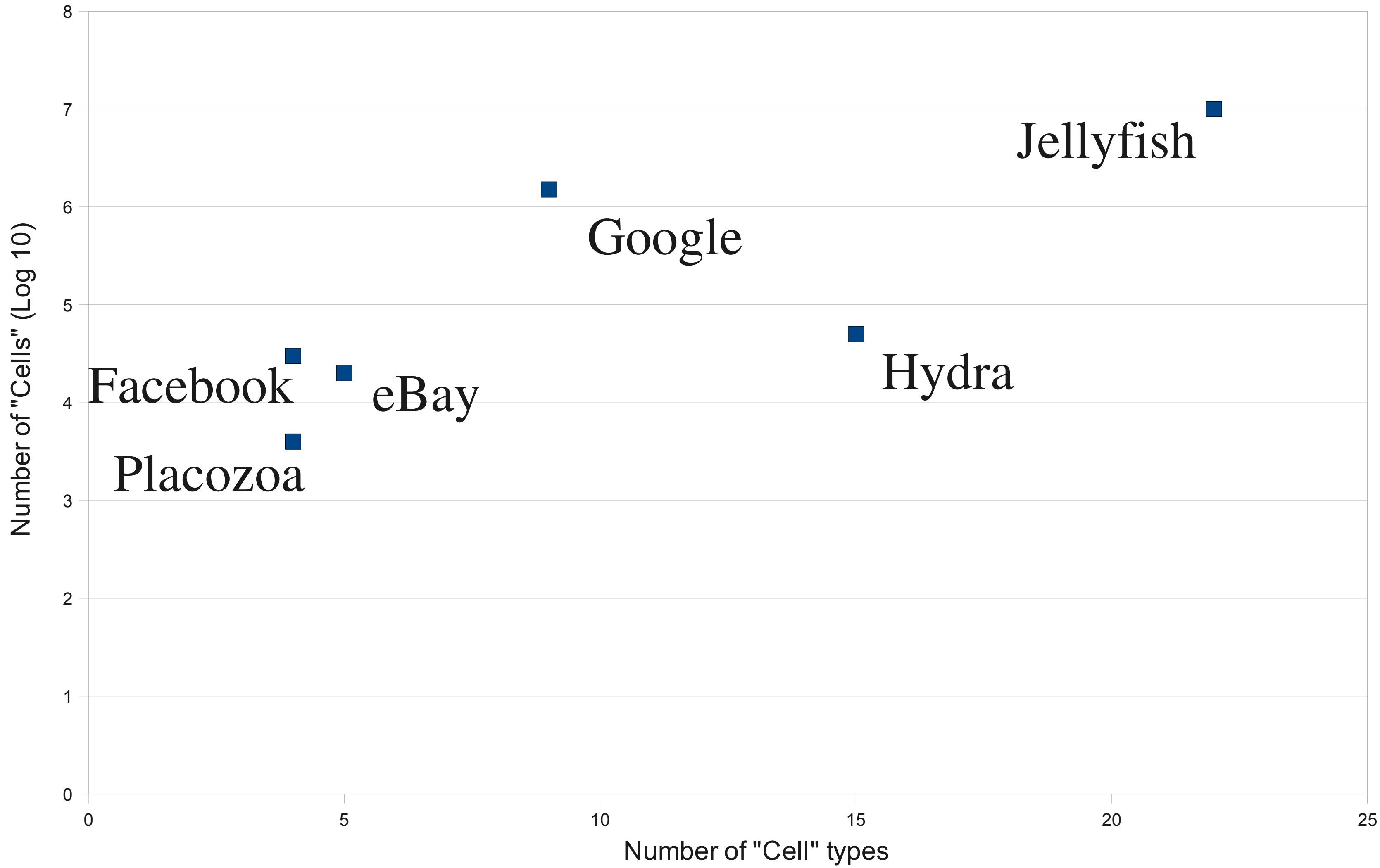
- Hydra
  - Estimated number of cells 50,000 - 70,000
  - At least 15 cell types



- Jellyfish (Cnideria Cyanea shown here)
  - Number of cells > 10 million
  - At least 22 cell types including neurons, sensors, muscle, endocrine, and Nematocysts (stinging cells) used for capturing prey.



Multicellular Complexity
Number of “Cells” vs. Number of “Cell” Types



Four Architectural Prerequisites for Multicellularity

Messaging – in multicellular transition, single cells forego direct DNA transfer in favor of polymorphic protein messages.

Apoptosis (or Programmed Cell Death) – controlled suicide of individual cells for the benefit or protection of the whole organism.

Differentiation – permanent specialization of cells. In humans there are about 250 specialized types of cells.

Stigmergy – organization and orchestration via a shared physical structure that is created by the organisms that it helps to organize.

Messaging and Polymorphism

Bacteria communicate by “conjugation” – direct exchange of DNA (thin red thread at left). Multicellular organisms DO NOT exchange DNA. Loewenstein calls that rule, “…the taboo of intercellular transfer of genetic information.” See The Touchstone of Life, Werner Loewenstein, Oxford University Press, New York, 1999 [p. 277]

Multicellular organisms communicate by polymorphic messenger molecules, i.e., molecular messengers whose “meaning” is determined by the receiving cell, not the sending cell. This is echoed by message sending in Object Oriented programs or Service Oriented Architectures in the Internet.

Genes determine both sides (sending and receiving) of behavior, hence evolution selects for the whole orchestration around a given message.

Today, computers allow all too easy installation of code...in effect they “conjugate” with servers on the Internet. THAT SHOULD BE TABOO

Swarms: Orchestration via Real-Time Messages

Flocks of birds, swarms of insects, and schools of fish organize their group movements by real-time visual signals (and presumably also in part by air or water pressure waves)

Life is often orchestrated by real-time messages

- E.coli motor control messenger molecules (CheYp) are generated and consumed in real-time.
- Traffic patterns emerge in real-time by visual signals
- Human dancing is organized by auditory signals (with a little help from tactile and visual signals)
- Peer-to-peer file sharing is based on (near) real-time messaging between the participating computers.

However real-time messages tend not to persist long, and are not localized. This limits their usefulness in orchestrating action in the 3-D world.

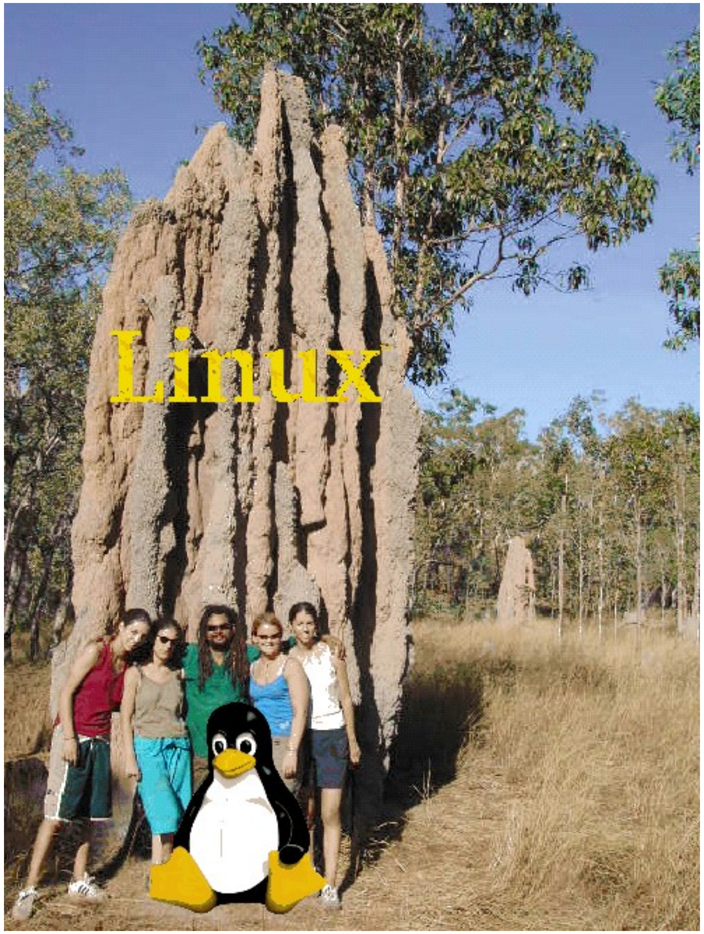
Longer Lasting than Swarms: Orchestration via Stigmergy

- ***Stigmergy*** refers to organization that is based upon structures external to the living cells/organisms that are built by the very creatures that inhabit them. These structures act as a persistent repositories of messages/data that helps to organize behavior over time and space.
- The term was coined to describe organization of social insects (ants, termites, bees) that build nests, mounds and hives that thereafter help to organize their activity.
- Both the 3D structure and the behavioral organization of messages attached to the stigmergy structure provide the locus of the “self” of the cells, or the organism or the colony
- Examples in life
    - The cytoskeleton of individual cells
    - The extracellular matrix of multicellular organisms
    - Trunks, branches and leaves of plants
    - The physical “nests” built by termite, ants and bees is a stigmergy structure (by definition)
    - Buildings and Cities are stigmergy structures for human societies
- Ant societies based on persistent cues (stigmergy) are more complex than ant societies based on real-time signal messages (Anderson & McShea, 2001)

Orchestration via Network Stigmergy Structures

- The topology inherent in the stigmergy structure has strong implications for the topology of the organization it supports. Structures in 1, 2, or 3D space, e.g., ant trails bee honeycombs, or termite mounds, support 1, 2, or 3D organization of behavior.
- Brains, or ganglia, of networked neurons in biological organisms are logically complex networks even though they are embedded in 3-D space. Sensory information is typically mapped onto the 2-D surface of the cortex. And much filtering and associative processing occurs in 3-D. However the very complex web of synaptic connections goes far beyond 3 spatial dimensions.
- Point-to-point addressability in the Internet, e.g., with IP addresses, is akin to neural synaptic messaging. It is possible (but not desirable) for every "cell" in a multicellular computer to have network communication functions similar to neurons.
- A stigmergy structure built on a network topology is ideal for modeling other sorts of network
    - Google, for example, models the Web
    - Facebook, twitter, etc., model social networks
    - A complete Semantic Net (Sowa 1976, 1984, 2000) would be a worthy task for a serious multicellular computer – or a brain!

Stigmergy in Computing



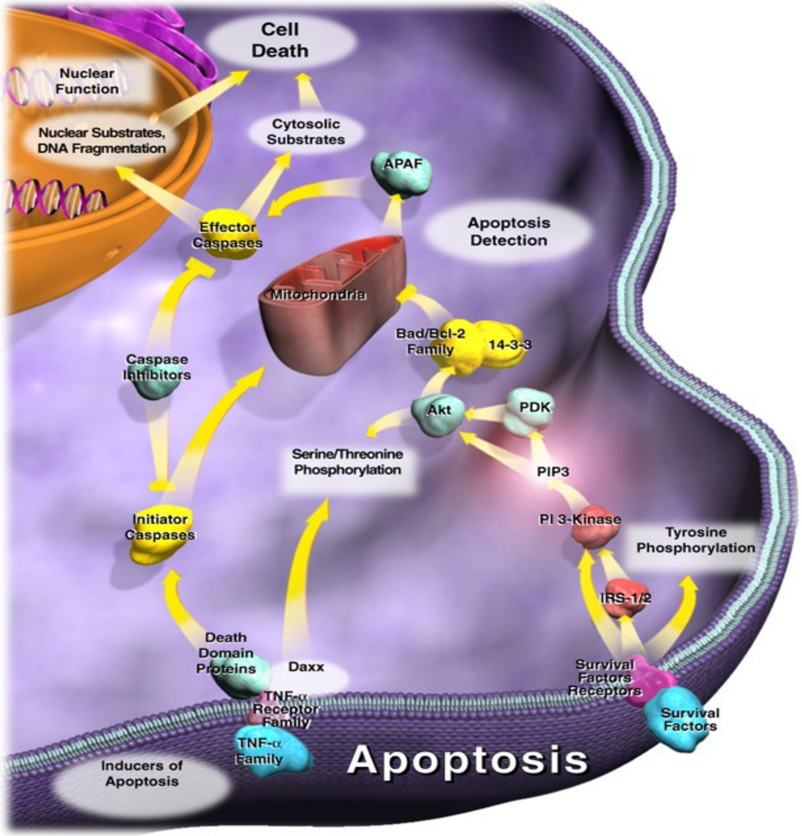
- Disks and Windows registry provide coordination for individual computers
- “Blackboard” systems coordinate between agents.
- Databases serve as stigmergy structures for companies and communities.
- DNS (the root server and caching servers)
- GPS (the structure is a set of satellites that are modified by the GPS groundstation and read by all GPS devices)
- Linux is a self-organized stigmergy system in the Internet
    - The persistent structure is provided by the CVS code repository

Self/Non-Self in Multicellular Organisms

- The Immune system is not the primary determinant of “self.” Immune systems evolved after multicellular life evolved.
- An immune system analog in computing that defines “self” cell-by-cell according to its “identity” will always be subject to spoofing, false positives, and false negatives.
  - Autoimmune diseases are false positives
  - Stealth infections, HIV for example, are false negatives
  - Cyber criminals will spare no effort to exploit those weaknesses. Note that hidden rootkits are similar to hidden HIV viruses, and just as deadly.
- Proper association with the stigmergy structure, not the genetic identity of cells, determines “self”
  - Half of cells in a human are bacteria or other non human cells.
  - Human cells that are not properly attached to the stigmergy structure are dangerous to the organism (a hallmark of cancer). Note that blood cells are specialized to not require attachment. They are exceptions to the rule.
  - Certain stigmergic messages (i.e., messages attached to the body's stigmergy structure) are interpreted by healthy cells as “You're OK” messages. When those messages are absent due to detachment, the cell determines that it is not OK, and commits suicide.

Apoptosis (Programmed Cell Death)

- A message API for controlling a carefully choreographed process programmed into each and every cell in a multicellular organism.
- Apoptosis is a prerequisite for multicellular life
  - Apoptosis evolved coincident with the first types of multicellular life - bacterial biofilms.
  - It evolved to deal with the sorts of issues that plague multicellular organisms but not single cell organisms such as replicative DNA damage, viral infection, and cells “going rogue.”
  - It solves those issues from a multicellular perspective - sacrificing the individual cell for the good of the multicellular organism.



The Role of Apoptosis

- It is not simply to cause dangerous cells to suicide in a safe way
- It also is used to “sculpt” the developing organism
  - It is central to the major changes in insect bodies as they change from larvae to pupae to adult
  - It is how tadpoles lose their tails
  - It is how human fingers and toes separate from each other
  - And it is crucial to the developing human brain. About half of all developing neurons fail to get enough positive reinforcement from their target neurons to avoid apoptosis. That is, twice as many are created than are needed, and those that don't turn out to make valuable connections commit suicide. (a useful model for multicellular computation?)
- Between 50 and 70 billion cells die each day due to apoptosis in the average human adult.

Apoptosis in Computing

Central lessons of apoptosis

- The system must be architected so that no cell is indispensable
- Every computer in the system should be responsible for monitoring its own “health” and shutting down or detaching from the network upon detecting its own anomalous behavior.
- If reliable apoptosis is a base part of the architecture, multicellular systems can spawn various specialized servers safe in the knowledge that those not rewarded for useful work can be set up to recognize that fact and remove themselves, making way for other types of server (that is, ask for a new software reload).

The Four Principles are Intertwined

	Facilitated By	Necessary Because
Specialization	A homeostatic environment is maintained by messaging within the whole organism. Cells rely on it for general tasks	Unspecialized cells are overly complex and have incompatible or conflicting functions
Messaging	Cell-surface receptors are specialized by cell type.	The receiving cell must determine the "meaning" of a message and not respond to messages not relevant to it
Stigmergy	Exportable messages survive outside the cells and the extracellular matrix provides attachment locations for long-lived messages	Extracellular matrix provides physical organization and relative location information and defines boundaries of "self"
Apoptosis	"I'm OK, you're OK" messages, are attached to stigmergy structures.	Organism is more important than any cell. A cell detached from its proper location is a danger to the organism.

Last (But not Least) the Multicellular "Sensorium"

Multicellular life increases the scale and accuracy of sensory input and the scale of effector behavior, i.e., the effectiveness of the organism's "awareness" and manipulation of its environment.

- In most cases, this is the very point of Information Processing
- And it's not a bad definition of Darwinian "Fitness" (a tricky topic to be sure)

Single cells have wide range of "senses" but multi-cellular organisms can coordinate inputs from many cells giving better detail and dynamic range.

The senses available to coordinated individual computing "cells" (e.g., iPhones) will determine what information can be organized by higher levels. iPhones directly "sense", i.e., capture, text, audio, image, video, location (GPS), and (3-D) acceleration.

- For example, "images" from many Flickr accounts or location information from many GPS devices (cell phones) can be combined to provide a coordinated view of the world.
- Ask yourselves: "How will one multicellular computing system sense another?"

The Multicellular “Designer Perspective”

- Let's return to preparing for the arrival of the shipping containers
- How do you even think about the problem?
- What do you want your multicellular computer to be when it grows up?
- What will the stigmergy structure(s) need to be?
- What specialized types of servers will there be?
- What messages will be acted upon, i.e., what are the APIs, or the virtual machine interface(s)?
- How will the servers recognize failure or infection by viruses, worms, backdoors, rootkits, etc.? And, if infected, what will they do?
- And, of course, how will you “program” the various servers, load them with images, and maintain them. That is, what sort of software development and maintenance tools do you need to create?

OR, what would Google do?

It's the Beginning, Not the End

Questions and Discussion