WOA 2008
Palermo

Towards
Eternally Adaptive Service Ecosystems

Franco Zambonelli

Agents and Pervasive Computing Group

Università di Modena e Reggio Emilia

Outline

- Motivations and related areas
  - Autonomic computing & communications
  - Cyber-physical systems
- A reference architecture
  - World, laws, species
- Metaphors
  - Physical, chemical, biological, ecological
- Pros and cons
- Research agenda

Motivations (1)

- Several emerging trends in computing and communication
- Service models and service provisioning are changing
  - High dynamics and variations of demands
  - Need for 24/7 availability and reliability
- Networks are changing
  - Convergence of Internet and Telecommunication networks
  - High dynamisms and decentralization
  - Integration with personal, mobile, and pervasive devices
- And so management needs are changing
  - Requires self-management and self-configuration
  - Humans "out of the loop"
  - 24/7 availability at zero human costs

The "Autonomic" Trend

- Mostly industry-driven research initiatives
- "Autonomic"
  - The term is borrowed from the "autonomic nervous system"
- Related to the idea of:
  - Giving modern ICT systems a sort of "nervous system"
  - Capable of reacting to contingencies and of regulating in autonomy the overall metabolism of such systems
  - Metabolism = Functional and Non-functional behaviours
  - i.e., self-management, self-adaptation, self-organization, self-healing, self-configuration, etc. "self-*" features

Autonomic Computing vs Autonomic Communication

- Two different perspectives on trying to embed self-* features in modern ICT systems
- Autonomic Computing:
  - Focus on resource management and reliability (large data- and service centres, large service systems)
  - Main drivers: IBM, Intel, HP
- Autonomic Communication:
  - Focus on network dynamics and network reliability (network management, mobile networks, pervasive networks)
  - Main drivers: Telecoms, Consumer Electronics
- In any case, the distinction between the two is sometimes "fuzzy"

The IBM Autonomic Computing Initiative

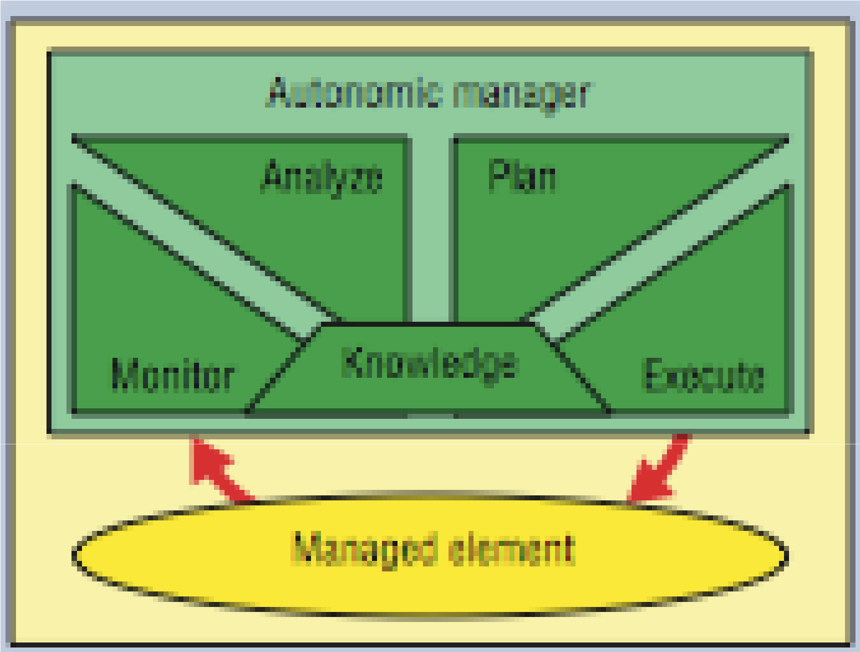
- Manifesto Launched in 2005
  - Motivated by the need to reduce the costs related to the configuration, optimization, healing, protection, of large ICT systems → moving humans out of the loop
  - Clearly, all large-scale and complex software systems shares the same goal (e.g., large-scale mission-critical systems)
- Quoting from the IBM manifesto:

Table 1. Four aspects of self-management as they are now and would be with autonomic computing.

Concept	Current computing	Autonomic computing
Self-configuration	Corporate data centers have multiple vendors and platforms. Installing, configuring, and integrating systems is time consuming and error prone.	Automated configuration of components and systems follows high-level policies. Rest of system adjusts automatically and seamlessly.
Self-optimization	Systems have hundreds of manually set, nonlinear tuning parameters, and their number increases with each release.	Components and systems continually seek opportunities to improve their own performance and efficiency.
Self-healing	Problem determination in large, complex systems can take a team of programmers weeks.	System automatically detects, diagnoses, and repairs localized software and hardware problems.
Self-protection	Detection of and recovery from attacks and cascading failures is manual.	System automatically defends against malicious attacks or cascading failures. It uses early warning to anticipate and prevent systemwide failures.

The MAPE-K Model

- The key component suggested by IBM to achieve autonomicity is the so called "Mape-K" one
  - An element of a system is coupled with an "autonomic manager"
  - Devoted to Monitor, Analyse, Plan, Execute, based on Knowledge
  - Such that the managed component is made "autonomic"
- Directly inspired by goal-oriented agent architecture,
  - but with a more explicit "close control loop"



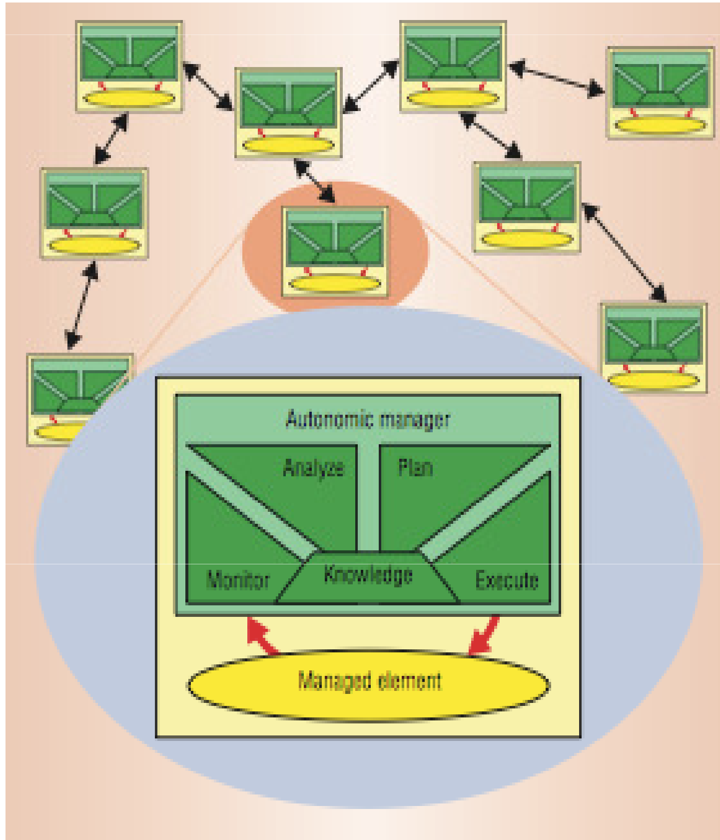
Elements of the MAPE-K Model

- Monitor
  - Gather information about the current behaviour of the component (e.g., response time, resources exploited, number of requests, etc.)
  - Tools: traditional, and also more "cognitive" monitoring tools
- Analyse
  - Try to understand what is happening (e.g., is fine? Are there performance problems? Are there security problems? Are there faults?)
  - Which of course requires "Knowledge", the capability of understanding data
  - Tools: classifiers, probabilsitic reasoning, ontologies
- Plan
  - Decide corrective actions in the case of problems (e.g., gather more resources, adopt an alternate class, increase priority of execution, re-boot, etc.)
  - Tools: logic-based models, BDI, policy-oriented, etc.
- Execute such actions on the managed element

Distributed MAPE-K System

WOA 2008
Palermo

- In case the system is made up of multiple (possibly distributed) elements
  - And this is the case for many data centres, service centres, service systems
- One can think at having the different autonomic managers cooperate with each other
  - Recognition of problems involving more than one entity
  - Distributed agreement on remedial actions



Advantages and Disadvantages of the MAPE-K Approach

- Advantages
  - Simple and clean model (clear control loop)
  - Can be applied to existing systems in the form of a separate "control plane" (in theory, but the practice is more difficult)
  - This is why it has become a sort of "reference approach" and it getting increasingly applied
- Disadvantages
  - Autonomic capabilities are not "inherent" in a system, but reside on a separate control plane (this is not good for the long term)
  - There is not real self-organization and self-adaptation in the system
  - Heavy weight, hard to be applied in modern Telecom and (pervasive) networks scenarios

Motivations (2)

(as pertaining to Situated and Autonomic Communications)

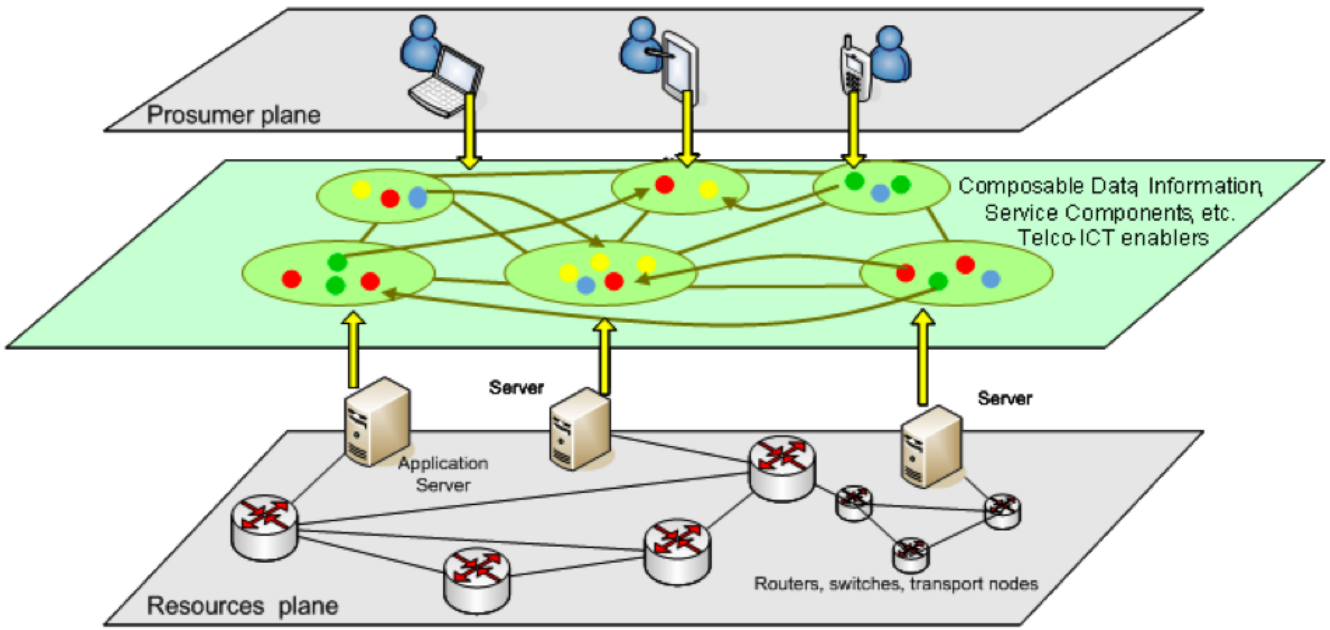
- Other than the need to integrated Self-* features in network and network/Telecom services to increase reliability and reduce management costs (a shared goal with Autonomic Computing)
- Convergence of Telecom and Internet scenarios
  - Need of a unifying light-weight approach and infrastructure
- Need for decentralization
  - Not only big service/data centers and no vertical integration
- Increased dynamics and scale of network scenarios
  - Mobile nodes, sensors, users,…, billions of nodes involved
- All of the above aimed at
  - Provide better and more flexible and diverse communication services to users
  - At reduced costs and thus with increased revenues
- Clearly, all network systems would share this…

The EU Initiative on (Situated and) Autonomic Communications

- Consultations started in 2005 with the strong support of EU Telecoms and of Network companies
  - Initiative Launched in 2006
  - Now absorbed into the "Internet of the Future" initiatives
- Key Goal:
  - Re-thinking Network architectures and services
- Quoting from the ICT Workprogramme:
  - "The goal of this initiative is to promote research in the area of **new paradigms** for communication/networking systems that can be characterised as situated (i.e. **reacting locally on environment and context changes**), **autonomously controlled, self-organising**, radically distributed, technology independent and scale-free. Consequently, communication/networking should become task- and knowledge-driven and fully scalable".

The Overall Approach to Autonomic Communication

- Mostly layer-less network architecture
  - All components (devices, users, producers, network agents, etc.) part of the same open "P2P plane"
  - Interacting/coordinating/linking aggregating with each other so as to dynamically self-organize and self-adapt services and functionalities
  - In a fully decentralized and unmanaged way



Key Elements of Autonomic Communication Approaches

- Structure of components
  - Mostly reactive (but more complex autonomous agent-like components are not excluded)
  - Capable of moving in the network and/or of diffusing simple signals around
- Structure of interactions
  - Biologically and/or Socially inspired
  - Slime-mold aggregation of resources, ant-based routing of packets and information, firefly synchronization, gossip-based broadcasting, etc.
- Structure of the environment
  - Contextual-knowledge, knowledge network plane
- Somewhat we can see it as a proper mixing of P2P approaches with bio-inspired approached
  - Components interact in a P2P way
  - But according to bio-inspired algorithms

Autonomic Communications and Middleware

- What is middleware in autonomic communications?
  - There are no "hardwired" general-purpose middleware services
  - All middle-level services are dynamically composed within the same P2P plane
- Deconstruction of the middleware concept
  - Network services aggregate with all the needed components to achieve a specific goal
  - This can include diffusing information to discover components, recruit mates to support proper routing of information, etc.
- This also implies a blurred distinction between data components and service components

Autonomic Computing + Autonomic Communication

- Synergies are possible between the two approaches
- Adopt the "MAPE-K" model for individual components, whenever needed
  - These would be thus more "cleaver" than simple reactive agents
  - Would be capable of self-managing themselves independently of the rest of the works
- Have the various elements (even those based on MAPE-K managers) interact via P2P bio-inspired network
  - To exploit self-organization and self-adaptation at the network level

Advantages and Disadvantages of Autonomic Communication

- Advantages
  - Very clean and light-weight approach, very suitable for future network scenarios → the idea of a single plane for components is to be shared, and so it is the idea of "deconstructing" middleware and of blurring the distinction between data and services
  - Potential to open brand new possibilities for the effective management of complex network systems and adaptable network services
- Disadvantages
  - Too much focus on network services and few on user services
  - Investigation of one-of solutions rather than of very general ones
- Therefore
  - We expect the vision to be absorbed slowly, in the forms of specific nature-inspired solutions to specific problems
- Yet, there is need of something more…

Cyber-Physical Systems

- A vision mostly driven by producers of consumer electronics
  - With an eye on the final users other than on the network!
- Our future network and service systems will form complex clouds of
  - Self-managing components, simple reactive components, data components, sensing devices, personal devices
  - All of which have to seamlessly and spontaneously interact with each other and with the user
  - Without any configuration needs and in a reliable and personalizable way
- Strictly connected with the physical and social worlds
  - Capable of self-organizing their overall spatial activities in a autonomous way
  - Capable of adapting to user needs and to their social context
  - Evolving according to evolution of user needs and to evolution in the physical world…

Why Cyber-physical System?

- Well, the overall vision pave the way for many industrial actors to contribute with "small" products to the delivery of very effective services
  - Imagine what additional features I could get from my iPhone if I had the possibility of having it freely and autonomously interact with other devices, sensors, people
  - Imagine how attractive could by any even very simple gadget that could play some role in the system (and this is why producers of consumer electronics are very interested in that vision)
  - Imagine how "social" Web platform could become if being part of an overall system where data about everyday social activities of users could be continuously collected…
- However, this requires more than simply "self-*" features
  - It requires systems to be adaptive in the presence of diversity
  - It requires eternity

Motivations (3)

- Prosumption and personalization
  - Users also act as producers of data and services (prosumers)
  - Similarly for producers of consumer electronics
  - People don't want "a" network, but "own" networks
  - Context-awareness, location-based, and in general adaptive cyber-physical interactions
- Long tail
  - Companies need to cover an increasing diversity of possible services, data, and devices
  - Prosumers contribute to such diversity
- Eternal betas and eternal evolution
  - No service/software components is ever ultimate
  - New components gets on appearing

What do we need?

- We need innovative architectures and software infrastructures to accommodate the changing scenarios and to be as "eternal" as possible
- Current approaches are not adequate
  - Focus on limited set of self-* features
  - Limited support for prosumption, diversity, and evolution
  - Limited support for users and for "cyber-physical" aspects
- Most research proposals, though
  - Tend to investigate one-of solutions to specific problems of networks, services, data
  - Contrasting trade-offs between different solutions
  - The result could be a complex patch-works of services, models of service interactions, models of data, models of devices, which by no means ensure eternity and easy adaptivity

The key question

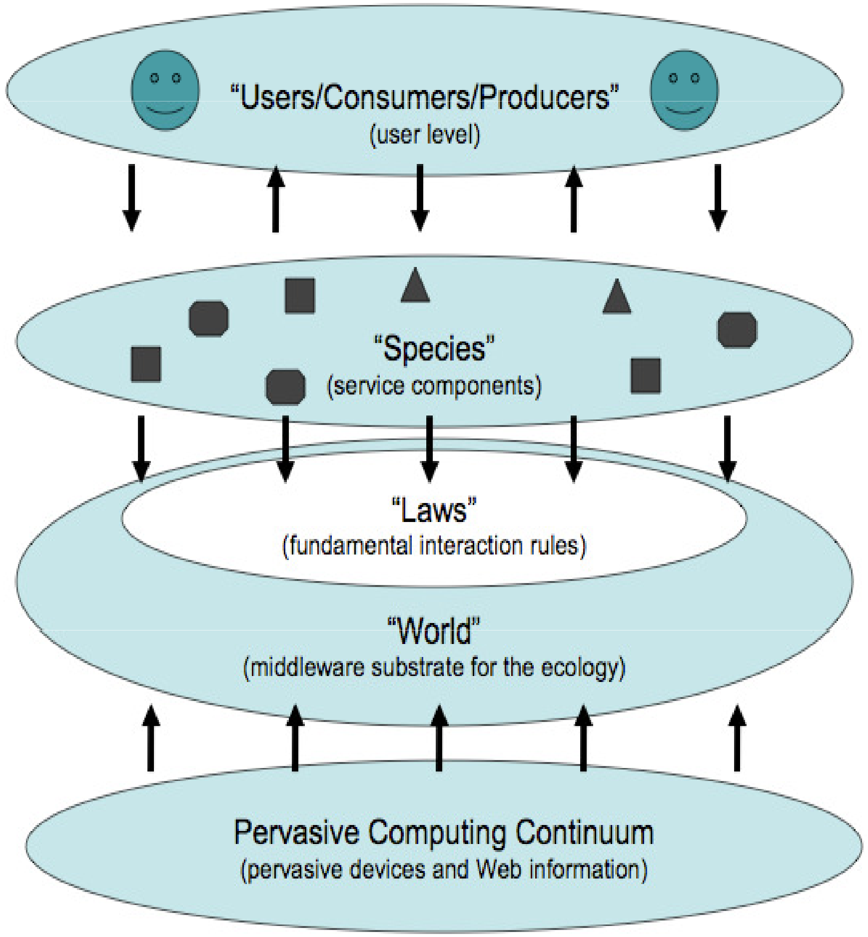
- *Is it possible to conceive a radically new way of modelling/building integrated network services and their execution environments, such that the apparently diverse issues of enabling autonomicity, pervasiveness, network-integration, context-awareness, dependability, openness, diversity, flexible and robust evolution, can all be uniformly addressed once and for all?*
- Need to re-think network, service, and data models and infrastructure from the foundation
  - Components should no longer be conceived as localized loci of functionalities/data, to be orchestrated based on some middleware or P2P services with some "self-* "features
  - Dynamics, diversity, adaptability, evolvability, should be inherent "rules of the game"

Towards Service Ecosystems

- In natural systems (and whether you think at physics, chemistry, biology, or ecology)
  - Self-adaptation, self-configuration, self-management, are inherent part of their everyday life and dynamics
  - The infrastructure (i.e., the laws of nature and the universe) is eternal and does not change
  - It naturally accommodates diversity
  - Although their components may not be eternal, systems eternally evolve (just think at life on earth)
- We can start start from this to
  - Build an "eternal" infrastructure for next generation networks and services
  - Have an "ecosystems" of eternal and eternally adaptive services (there included data management services, user service, network services, devices) live and evolve over it

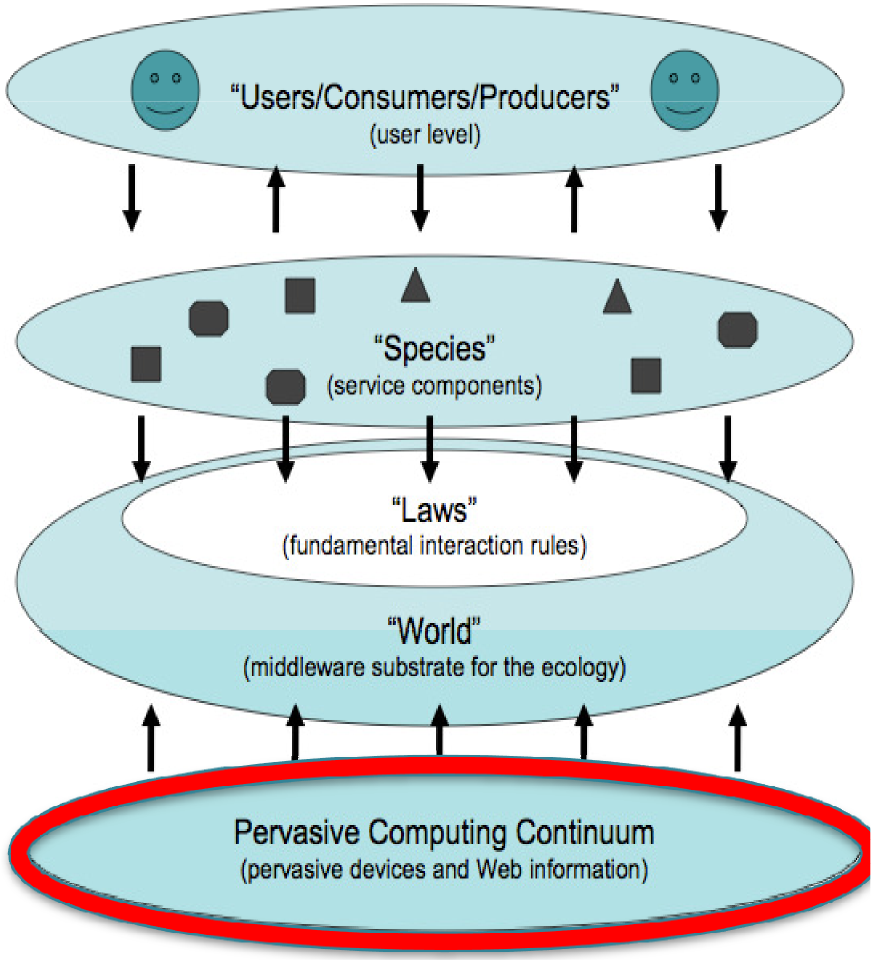
A Reference Architecture

- It abstracts from any specific nature-inspired metaphor
- Shows how general ecosystem concepts can be framed in a uniform way
- Useful conceptual guidelines to actually turn the conceptual architecture into an infrastructure for eternally adaptive service ecosystems



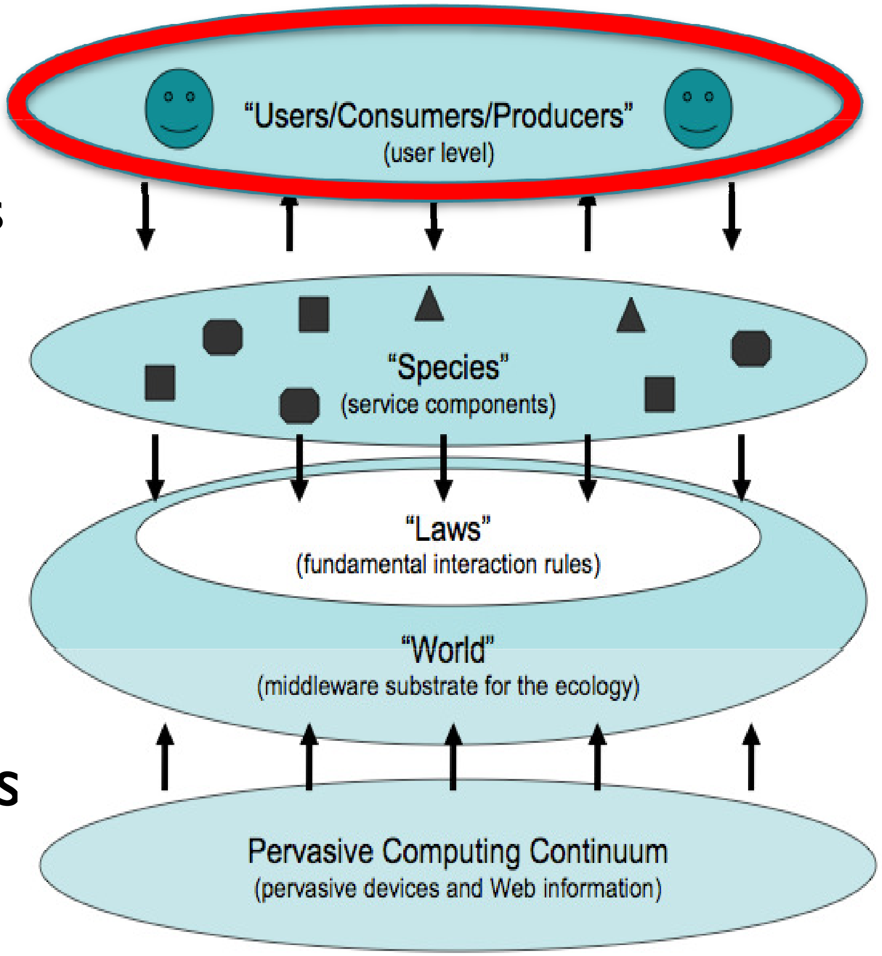
The Pervasive Computing Continuum

- Shaping the hardware ground on which the actual ecosystem will live and execute
  - Pervasive sensing and actuating devices very densely deployed in space
  - Personal computer-based systems
  - Wireless communications
- Feeding the ecosystem with data about nearly every facts of the world
  - Also via Web information



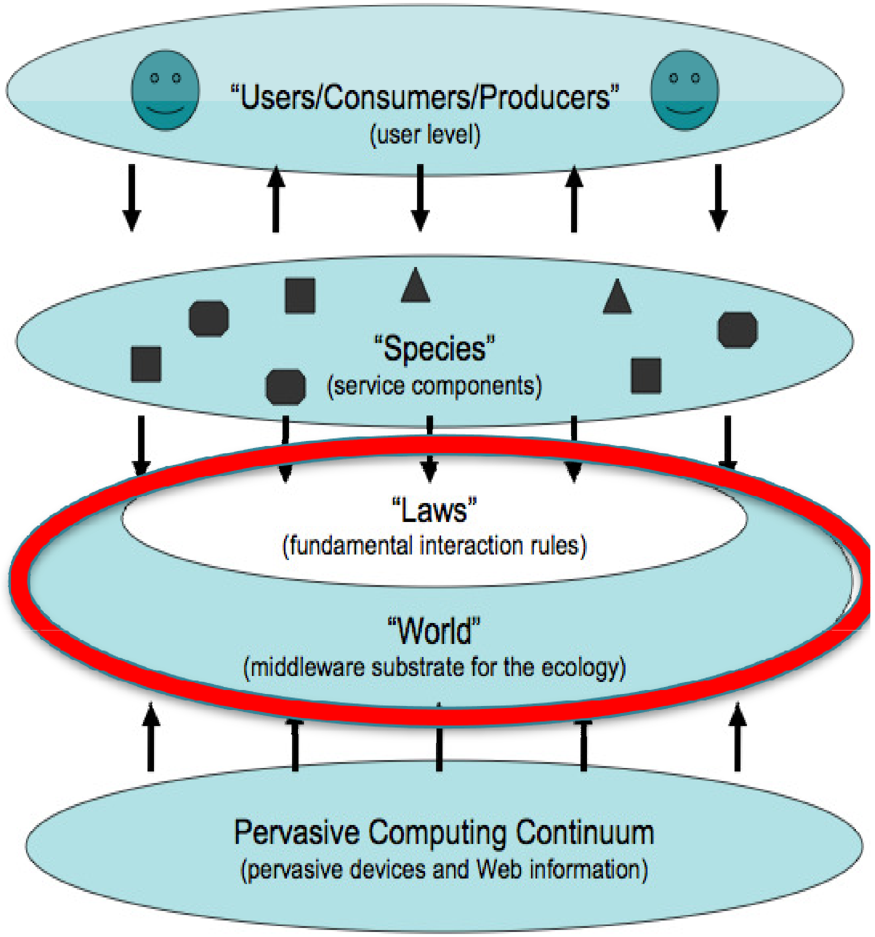
Users, Consumers, and Prosumers

- They can "*observe*", i.e., query, the ecosystem and its components
  - To obtain data, or computations
- They can "*extract*" components from the ecosystem
  - To consume data and service
- They can "*inject*" new components, data, and devices
  - To personalize the network
  - To deliver own services
  - To enforce control



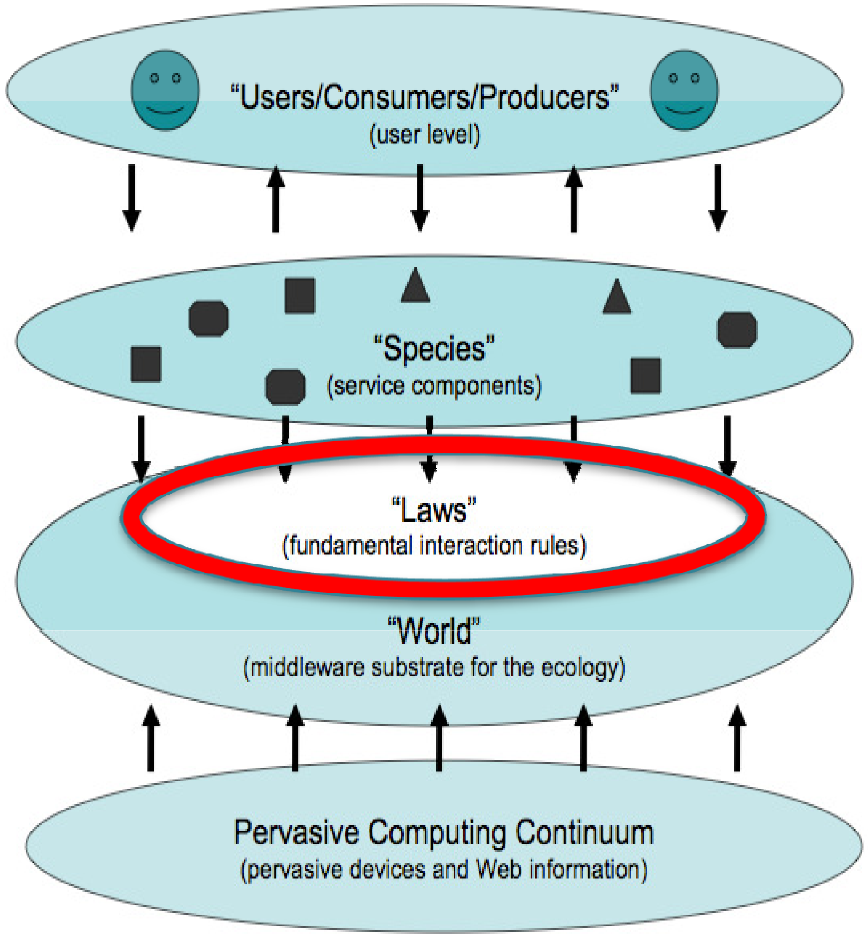
The World

- A very minimal middleware substrate
  - No "smart" middleware services
- Key goals
  - Supporting the lifecycle of components over a possibly dynamic and heterogeneous substrate
  - Enabling and enforcing interactions across components
  - According to the "laws of nature" of the ecosystem



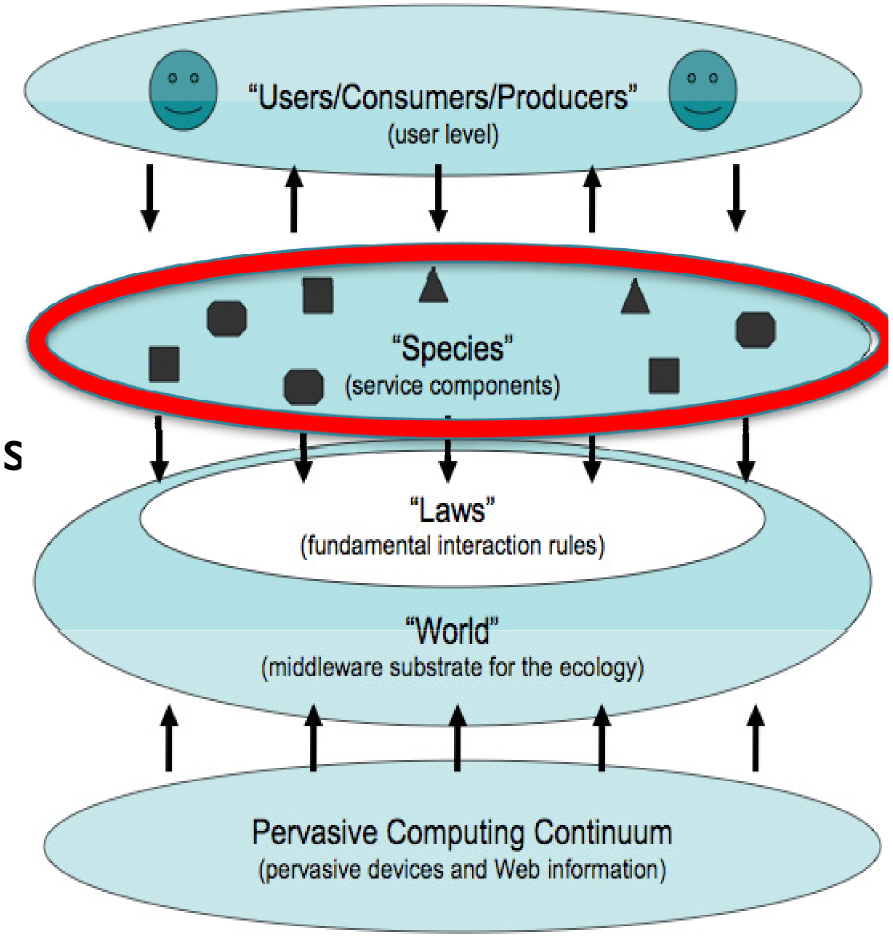
The Laws

- Ruling interactions and the overall dynamics and self-* behaviour of the system
  - How components should interact and when
  - How components should compose/aggregate
  - When component should die/clone/reproduce
- They are eternal
  - Species of components can change, laws can't
  - Laws apply to all components
  - Different species may react to laws in differentiated ways



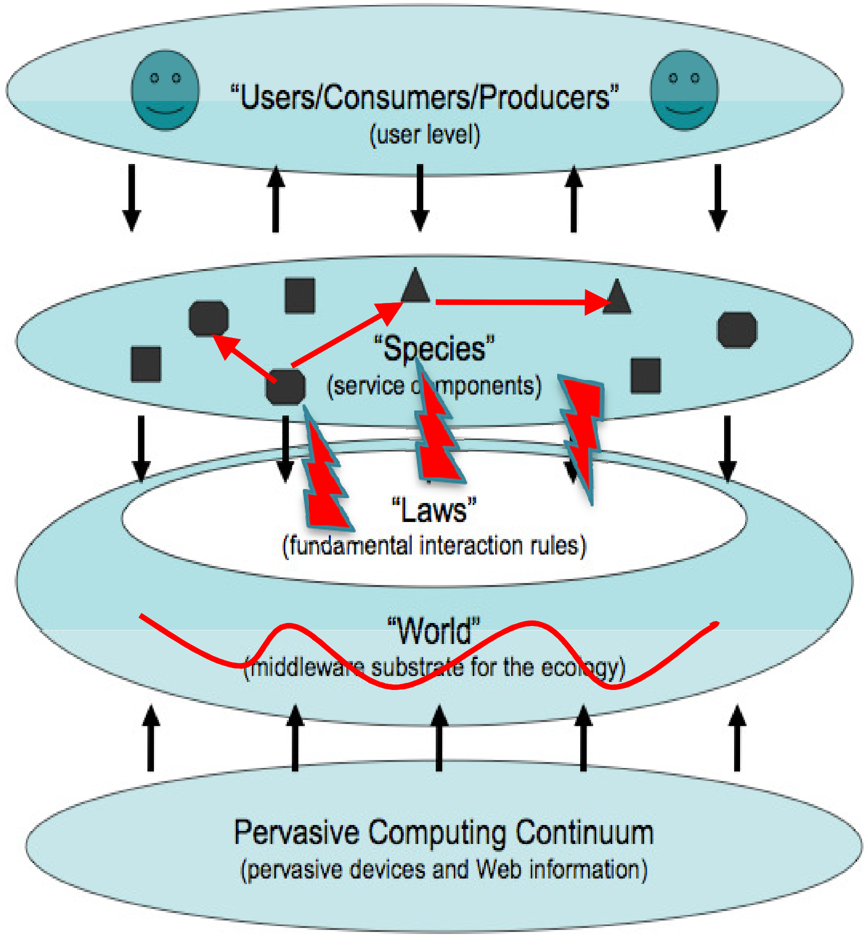
Species

- The software/digital components of the ecosystem
- May be of different nature and classes
  - "Passive" data items
  - "Active" computational entities
  - Interfaces to devices and Web
  - Can be dynamically injected
- Are all subject to the laws
  - But different components can react differently to laws
  - Based on internal characteritics and external interfaces



The Ecosystem Dynamics

- Species
  - Living in a region of "World"
  - Moving, acting, composing, as determined by laws
- Laws
  - Are typically local and impact on the local activities and interactions
  - The way they apply determined by the state of local components (feedback loop)
- World
  - The shape of space influence (and is influenced by) the above



Why Metaphors?

- Beside the abstract reference architecture of the ecosystem
  - How should its components, laws, world, be modelled?
  - What form should they actually take in implementation terms?
- Several possible natural metaphors can be adopted
  - Corresponding at different "levels of observation" of natural systems
  - Based on different mechanisms for laws and on different components behaviours
  - And in all of which self-* features, adaptability, and evolvability, are (to different extents) inherently expressed
- It is worth outlining that such metaphors, so far, have been mostly exploited for specific solutions, applications, and/or algorithms (e.g., in autonomic communications research) but never as a comprehensive approach

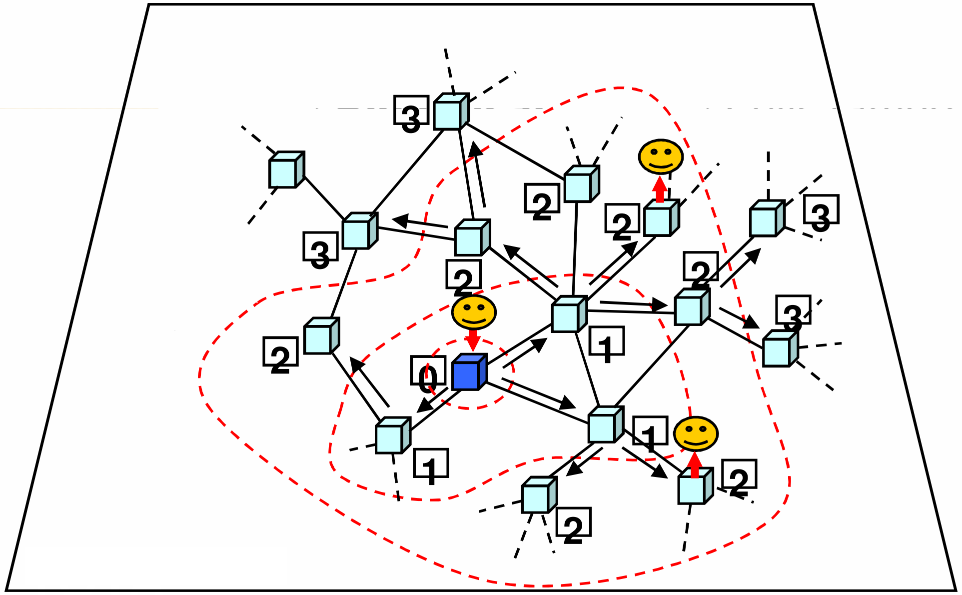
Metaphors

	Species	**Laws**	**World**
Physical	Particles (computational components) and messages (computational fields)	Navigation and activities driven by fields (gradient ascent by components)	The Universe (a network), as shaped by waves and particles.
Chemical	Atoms (semantically described) and Molecules (composed semantic descriptions)	Chemical Reactions (matching of semantic descriptions and bonding of components)	Space (localities/bags of components)
Biological	Cells (amorphous computing cells, modules of self-assembly components)	Diffusion of chemical gradients and morphogens, differentiation of behaviour and activity	Space (Abstract computational landscapes, or physical landscapes)
Ecological	Organisms (Agents) and Species (Classes) and Resources (Data)	Survive (goal-orientation), eat, produce, and reproduce	Niches (Pervasive computing environments

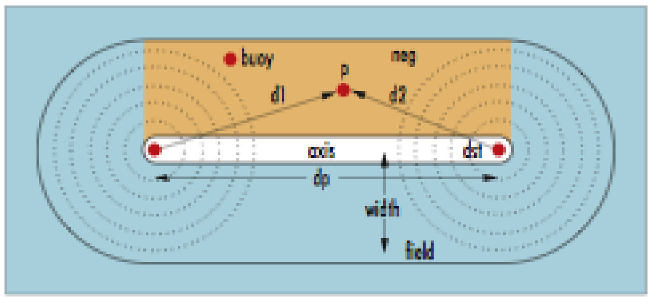
Metaphors

	Species	Laws	World
Physical	Particles (computational components) and messages (computational fields)	Navigation and activities driven by fields (gradient ascent by components)	The Universe (a network), as shaped by waves and particles.
Chemical	…oms …mantically …ribed) and	Chemical Reactions (matching of semantic descriptions and bonding of	Space (localities/…gs of compone…
	…ous …ls, f-		
	…es)	reproduce	

The TOTA Middleware (Mamei Zambonelli, 2004)

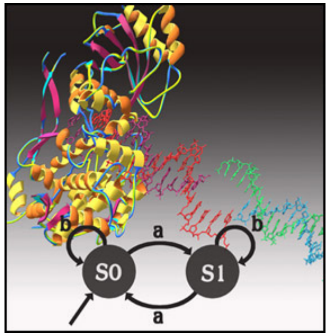


The Proto Language (Beal, 2004)



Metaphors

	Species	Laws	World
Physical	Particles (computational components) and messages (computational fields)	Navigation and activities driven by fields (gradient ascent by components)	The Universe (a network), as shaped by waves and particles.
Chemical	Atoms (semantically described) and Molecules (composed semantic descriptions)	Chemical Reactions (matching of semantic descriptions and bonding of components)	Space (localities/bags of components)
Biological	C... amorphous ...ing cells, ...of self- ... (Data)	Diffusion of chemical gradients and morpho... differen... behavio... activity	Space (Abstract computational landsc...
		Survive orientat... produce, and reproduce	...environments



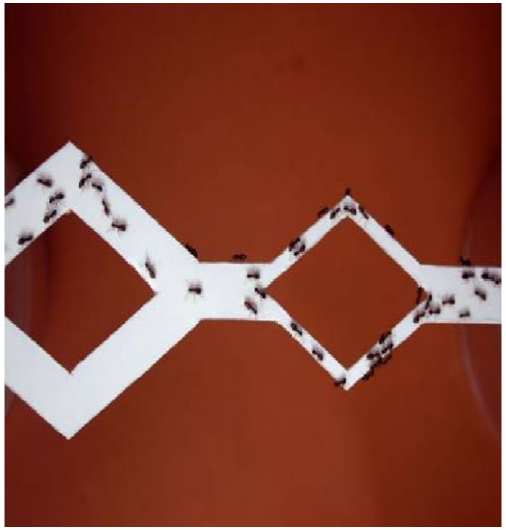
Computational Biology and DNA Computing

The **Gamma** Coordination Language (Banatre, 1990)

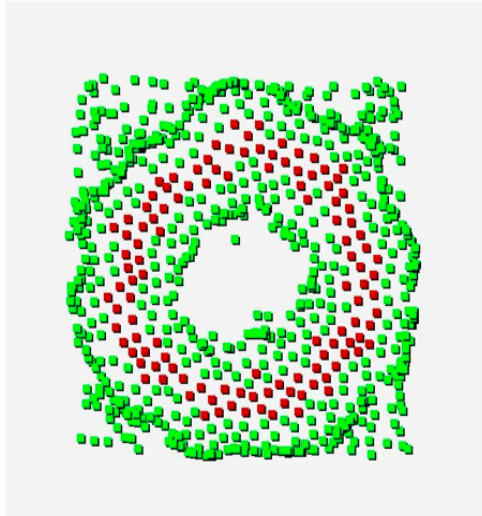
WOA 2008
Palermo

Metaphors

Ant Colonies (Parunak 2007; Babaoglu 2006)



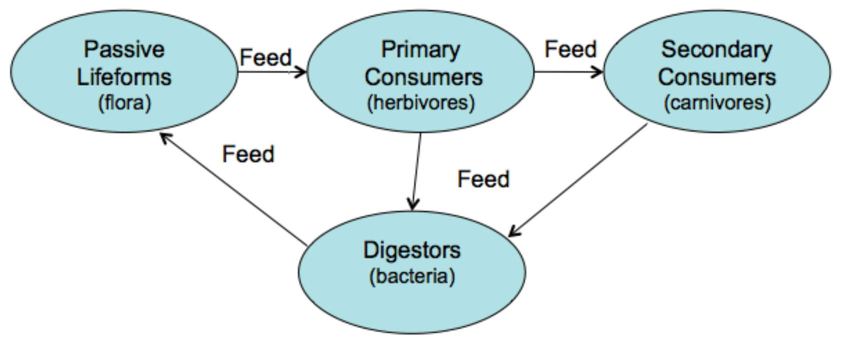
Amorphous Computing And Swarm Robotics (Nagpal, 2002)



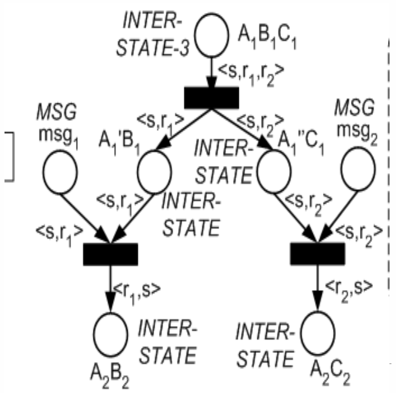
	(composed semantic descriptions)	Na ac by as co Ch Re (m semantic description bonding of components)	
Biological	Cells (amorphous computing cells, modules of self-assembly components)	Diffusion of chemical gradients and morphogens, differentiation of behaviour and activity	Space (Abstract computational landscapes, or physical landscapes)
Ecological	Organisms (Agents) and Species (Classes) and Resources (Data)	Survive (goal-orientation), eat, produce, and reproduce	Niches (Pervasive computing environments

Metaphors

Trophic Networks
(Agha, 2008; Zambonelli 2008)



Computational Economies of agents
(Jennings, 2003)



	Species		
Physical	Particles (computation components) messages		
		s... d... and b... co... s)	
	...modules of self-assembly components)	Dif... of che... gra... s and mor... gens, diffe... ntiation of beha... our and activity	Space (Abstract computational landscapes, or physical landscapes)
Ecological	Organisms (Agents) and Species (Classes) and Resources (Data)	Survive (goal-orientation), eat, produce, and reproduce	Niches (Pervasive computing environments)

Metaphors

Other metaphors ?

	Species	Laws	World
Physical	Particles (computational components) and messages (computational fields)	Navigation and activities driven by fields (gradient ascent by components)	The Universe (a network), as shaped by waves and particles.
Chemical	Atoms (semantically described) and Molecules (composite semantic descriptions)	Chemical Reactions (matching of semantic descriptions and bonding of components)	Space (localities/bags of components)
Biological	Cells (amorphous computing cells or modules of self-assembly components)	Diffusion of chemical gradients and morphogens, differentiation of behaviour and activity	Space (Abstract computational landscapes, or physical landscapes)
Ecological	Organisms (Agents) and Species (Classes) and Resources (Data)	Survive (goal-orientation), eat, produce, and reproduce	Niches (Pervasive computing environments

Comparative Analysis

- Are these metaphor equally suitable to address the needs of future service ecosystems?
  - NO!
  - But it depends on what you want to achieve
- Let’s analyse according to three dimensions:
  - **Space**: the capability of facilitating self-organization and self-adaptation of functional distributed patterns of activity
  - **Time**: the capability of tolerating evolution and increasing diversity
  - **Control**: the capability of being easy to understand, design, and control in a decentralized way

Pros and Cons

	Space (self-organization)	*Time (evolution and adaptation)*	*Control (decentralized management)*
Physical	+ (global self-organizing spatial structures)	-- (no new components, always same behaviours)	++ (we know well how to build and control specific structures in physic)
Chemical	+ (mostly local self-organizing structures, sometimes global too, as in crystals)	++ (several new components can be generated under the same basic laws)	+ (reactants and catalysts can exert control over the dynamics and structure of reactions)
Biological	+ (local, morphogenesis of local shapes)	-- (limited number of new "shapes", and only local changes)	- (mechanisms of morphogenesis not fully understood)
Ecological	+ (local structures mostly, although sometimes leading to more global patterns)	++ (several new species and same laws)	-- (difficult to understand how to enforce control over ecologies of many species, at most only local centralized control)

Summary

- Self-organization and self-adaptation
  - Physical and biological metaphors are the most well understood and extensively studied in several computational scenarios
  - Chemical and ecological metaphors could work equally well
- Diversity and evolution
  - Physical and biological metaphors are not directly suitable
  - Chemical and biological metaphors inherently accommodate them
- Control
  - Physical and chemical systems are well understood, and tools exists to control them
  - Less control and understanding of biological and ecological dynamics

Extents of Applicability

- Small-scale and special-purpose systems and services
  - Diversity and evolution are not big issues
  - Physical, chemical, and biological modeling can be OK
- Eternally adaptive service ecosystems
  - Chemical or ecological modeling can be needed to accommodate diversity and evolution
  - Chemical may be quite too level and fine-grained
  - Ecological can be difficult to understand, model, and control
- We should look for some "hybrid" synthesis

My Own Research Agenda
(open research themes at my research group)

- We have extensively studies physically-inspired models in the past
- We are currently experiencing with
  - Ecological models based on trophic networks
  - Chemical models based on semantic self-composition of services (in cooperation with Univ. Bologna)
- The idea is to synthesize the two, bringing in there the lessons learnt in controlling physical systems
  - Simulations and experiments on a pervasive computing testbed
  - Case studies in the area of location-based services for adaptive people-to-people and people-to-environment coordination

General Research Questions

- Does what I have said make sense at all? If yes, then…
- Rather than getting inspiration from existing natural systems and laws, should we rather invent from stratch our own laws of nature?
- What about security in these kinds of scenarios?
- Can we actually implement this concepts in an effective and reliable way?
- Can we actually control these system?
- Will systems of this kind be ever accepted by industries and users?
- Can we accommodate legacy (i.e., can evolovability by achieved starting from the existing)?

Conclusions

- Nature-inspired service ecosystem have the potential to represent a sound approach to face, once and for all, several technical and social challenges for future and emerging network and service scenarios
    - i.e., for the realization of *eternally adaptive service ecosystems*
- However, there is still a lot of foundational and experimental research to do before even understanding if such an approach can be applicable and effective
    - Building on the lessons of autonomic computing and communication
    - And pushing them forward
- For sure, they are a source for a large variety of fascinating and fresh research questions!