AUTHOR QUERY FORM

ELSEVIER	**Journal: YJNCA** **Article Number: 616**	**Please e-mail or fax your responses and any corrections to:** **E-mail: corrections.eseo@elsevier.macipd.com** **Fax: +44 1392 285878**

Dear Author,

Please check your proof carefully and mark all corrections at the appropriate place in the proof (e.g., by using on-screen annotation in the PDF file) or compile them in a separate list.

For correction or revision of any artwork, please consult http://www.elsevier.com/artworkinstructions.

Any queries or remarks that have arisen during the processing of your manuscript are listed below and highlighted by flags in the proof. Click on the Q link to go to the location in the proof.

Location in article	**Query / Remark: click on the Q link to go** **Please insert your reply or correction at the corresponding line in the proof**
Q1	Please complete and update the reference given here (preferably with a DOI if the publication data are not known): Viroli and Zambonelli (to appear). For references to articles that are to be included in the same (special) issue, please add the words 'this issue' wherever this occurs in the list and, if appropriate, in the text.
Q2	Please check whether the designated corresponding author is correct, and amend if necessary.
Q3	Please check the telephone number and fax number of the corresponding author, and correct if necessary.
Q4	Please note that the reference style has been changed from a Numbered style to a Name-Date style as per the journal specifications.

Thank you for your assistance.

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences ☆

Cynthia Villalba, Franco Zambonelli *

Dipartimento di Scienze e Metodi dell'Ingegneria, Università di Modena e Reggio Emilia, Italy

ARTICLE INFO

Article history:
Received 13 July 2010
Received in revised form
19 November 2010
Accepted 1 December 2010

Keywords:
Pervasive computing
Multi-agent systems
Self-* systems

ABSTRACT

Pervasive and mobile computing devices increasingly populate our environments. In this context, innovative frameworks have to be identified for the deployment and execution of pervasive service systems made up of a massive number of components, and able to exhibit properties of *self-organization* and *self-adaptability*, and of *long-lasting evolvability*. This paper discusses how such frameworks could get inspiration from natural systems, by modeling and deploying services as autonomous agents, spatially situated in an ecosystem of other services, data sources, and pervasive devices, all of which acting, interacting, and evolving according to a limited set of "laws of nature". A conceptual architecture is introduced to frame the key concepts of nature-inspired approaches and to survey the key natural metaphors that can be adopted to realize the concept of pervasive service ecosystems. Following, the key characteristics of our original ecological approach are detailed, also with the help of representative case studies, and an extensive set of simulation experiments are reported to show the potential effectiveness of the approach.

1. Introduction

Pervasive and mobile computing devices increasingly populate our environments (Estrin et al., 2002; Want, 2006). These, together with the increasing amount of Web tools that make it possible to produce and access spatially situated information about the physical world (Castelli et al., 2007), will define a global-scale and very dense, decentralized infrastructure for general-purpose usage. At the user level, the infrastructure can be used to access innovative services for better perceiving/interacting with the physical world and for acting on it. It is also expected that users themselves will be able to personalize the infrastructure by deploying customized services over it. In addition, the infrastructure will be used as a way to enrich traditional classes of services with the capability of dynamically and autonomously adapting their behavior to the context in which they are exploited.

The effective development and execution of services in the above infrastructure calls for a deep re-thinking of current service models and of service frameworks, in order to: (i) Naturally match the spatial nature of the environment and of the services within, and rely on mostly localized spatial interaction to provide support for massive scalability (Zambonelli and Mamei, 2005; Beal and Bachrach, 2006; Locatelli et al., 2010). (ii) Inherently exhibit properties of self-organization, self-adaptation and self-management that are required in highly decentralized and highly dynamic scenarios (Dobson et al., 2006; Roy et al., 2008; Brazier et al., 2009). (iii) Flexibly tolerate evolutions of structure and usage over time (Zambonelli and Viroli, 2008). This is necessary to account for increasingly diverse and demanding needs of users as well as for technological evolution, without forcing significant re-engineering to incorporate innovations and changes.

To reach this goal, we should no longer conceive services and their interactions as in usual service-oriented architectures (Huhns and Singh, 2005), where services are simply functional entities orchestrated according to mostly static patterns and with the help of specific middleware services. No one can rely on ad-hoc one-of architectural solutions to achieve specific self-* features in existing systems, resulting in an increase of complexity (Kephart and Chess, 2003). Rather, the most promising direction is that of taking inspiration from natural systems (Mamei et al., 2006; Babaoglu et al., 2006), where spatial concepts, self-organization, self-management, and long-lasting evolvability are inherently there because of the basic "rules of the game".

We are aware that nature-inspired solutions have already been extensively exploited in the area of distributed computing for the implementation of specific middleware solutions or of specific

---

☆ *Previous publications*: This paper extends the paper published in the Proceedings of the Second Workshop on Bio-Inspired and Self-* Algorithms for Distributed Systems (BADS 2010). The extensions include: a critical survey of related approaches in Section 3; more details on the proposed original approach; an extended set of simulation experiments to more completely assess the proposed approach. Overall, the new paper includes more than 50% new material over the workshop version.

* Corresponding author. Tel.: +39 0 522 522215; fax: +39 0 522 522209.

E-mail addresses: cynthia.villalba@unimore.it (C. Villalba), franco.zambonelli@unimore.it (F. Zambonelli).

1084-8045/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2010.12.003

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

distributed services (Mamei et al., 2006; Babaoglu et al., 2006). Similarly, we are aware that natural and ecological metaphors have been adopted to characterize the complexity of modern ICT and service systems (Ulieru and Grobbelaar, 2007; Herold et al., 2008). Here, we go further than adopting natural system as an inspiration for specific algorithmic or middleware solutions or as an metaphor to characterize complexity. Rather, in this paper we argue that natural metaphors can act as a reference architecture around which to conceive, model, and develop a fully fledged pervasive service framework and all the components within.

Although one can think at different classes of natural systems and from different perspectives (e.g., physical, Mamei and Zambonelli, 2005; chemical, Viroli and Zambonelli, to appear; biological, Babaoglu et al., 2006; or social/ecological, Agha, 2008) one can always recognize the following characteristics: above a spatial environmental substrate, autonomous individuals (i.e., agents) of different kinds interact, compete, and combine with each other in respect of the basic laws of nature. Accordingly, in our scenario, the shared pervasive infrastructure substrate will have to be conceived as the space in which bringing to life an ecosystem of service agents, intended as individuals whose computational activities are subject to some basic laws of the ecosystem, and for which the dynamics of the ecosystem (as determined by the enactment of its laws) will provide for naturally enforcing features of self-organization, self-management, and evolvability.

In this context, the paper provides the following contributions:

- Introduces a reference architecture for nature-inspired pervasive service ecosystems, to show how ecosystem concepts can be framed into a unifying conceptual scheme (Section 2).
- Surveys the possible natural metaphors that can be adopted to realize nature-inspired ecosystems, also with reference to the state-of-the-art in the area, and discusses their pros and cons (Section 3).
- Details the specific ecological approach that we have started investigating. The approach abstracts the components of the ecosystem as sorts of goal-oriented organisms that driven by laws of survival, interact with each other and self-organize their activities according to dynamic food-web relations (Section 4). A representative case study (Ferscha, 2007) is also introduced to clarify the concepts expressed, and to compare with more traditional service-oriented architectures.
- We present a number of simulation experiments that we have performed to assess the effectiveness of our approach (Section 5). The experiments, simulating a spatial pervasive computing scenario, show that our approach is capable of effectively exhibiting interesting self-organization features in a variety of different situations.
- We describe other scenarios where our ecological approach can be effectively applied (Section 6).

Eventually, in Section 7, we conclude and outline open research directions.

2. A reference architecture for pervasive service ecosystems

A unifying conceptual reference architecture can be identified around which to frame the key abstractions and the conceptual structure for spatial pervasive service ecosystems, independently of the specific natural metaphor adopted (see Fig. 1).

At the lowest level is the physical ground on which the ecosystem will be deployed, i.e., a very dense and widely populated infrastructure (ideally, a world-wide pervasive continuum) of networked computing devices (e.g., PDAs, tags) and information sources (Web 2.0 fragments). At the highest level, service developers, producers and consumers of services and data, access the open service framework for using/consuming data or services, as well as for producing/deploying in the framework new services and new data components. At both levels, the architecture exhibits a high-degree of openness and dynamics, as new devices, users, services, data components can join and leave the system at any time. Between these levels, there are the components of the pervasive ecosystem architecture.

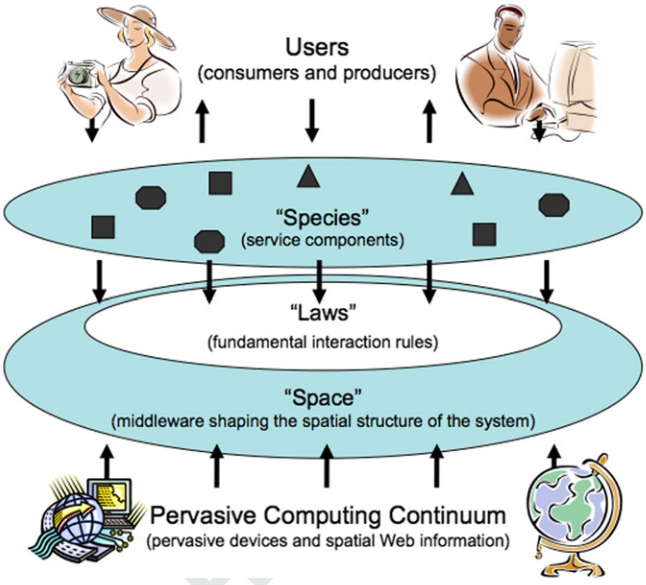

Fig. 1. A reference architecture for pervasive service ecosystems.

The "species" level is the one in which physical and virtual devices of the pervasive system, digital and network resources of any kind, persistent and temporary knowledge/data, contextual information, events and information requests, and of course software service components, are all abstracted as "living individuals" (or agents) of the system. Although such individuals are expected to be modeled (and computationally rendered) in a uniform way, they will have specific characteristics very different from each other, i.e., they will be of different "species". The dense population of devices and actors involved at the highest and lowest levels, together with their dynamics, reflect in the presence of a very massive and dynamically varying number of individuals and species.

The "space" level provides the spatial fabric for supporting individuals, their spatial activities and interactions, as well as their life-cycle. From a conceptual viewpoint, the "space" level gives shape to and defines the structure of the virtual world in which individual lives. What the actual structure and shape could be, might depend on the specific abstractions adopted for the modeling of the ecosystem. From a practical viewpoint, the spatial structure of the ecosystem will be implemented by means of some minimal middleware substrate supporting the execution and life cycle of individuals, and will enforce concepts of locality, local interactions, and mobility, coherently to a specific structure of the space.

The way in which individuals live and interact (which may include how they produce and diffuse information, how they move in the environment, how they self-compose and/or self-aggregate with each others, how they can spawn new individuals, and how they decay or die) is determined by the set of fundamental "laws" regulating the eternal service ecosystems model. Such laws, or "eco-laws", are expected to act on the basis of spatial locality principles, as in real laws of nature (which is also what makes real ecosystems scalable): the enactment of the laws on individuals will typically affect and be affected by the local space around them and by the other individuals on.

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

The dynamics of the ecosystem will be overall determined by having individuals acting based on their own internal goals, yet being subject to the eco-laws for their actions and interactions. The fact that the way eco-laws apply may be affected by the presence and state of other individuals, provides for closing the feedback loop which is a necessary characteristic to enable self-* features.

As far as adaptation over time and long-term evolution are concerned, the very existence of the eco-laws can make the overall ecosystem sort of eternal, and capable of tolerating dramatic changes in the structure and behavior of the species. Simply said in ecological terms: while the basic laws of life (i.e., the basic infrastructure and its laws) are eternal and do not change (i.e., do not require re-engineering), the forms under which it manifests continuously evolve (i.e., the actual service and data species), naturally inducing new dynamics for the interactions between individuals and for the ecosystem as a whole.

3. Survey of natural metaphors

The key difference in the possible approaches that can be undertaken towards the realization of eco-inspired service frameworks stands in the metaphor adopted to model the ecosystem, its individuals, the space in which they live, and its laws. All of the below surveyed metaphors inherently support, by construction, spatial concepts.

3.1. Physical metaphors

Physical metaphors consider that the species of the ecosystem are sort of computational particles, living in a world of other particles and virtual computational fields, the latter acting as the basic interaction means. There, all activities of particles are driven by laws that determine both how particles should spread fields, how such fields propagate and gets re-shaped upon changing conditions, and how particles should be influenced by the local gradients and shape of the computational field (those whose description "matches" some criterion). Particles can change their status based on the perceived fields, and they can move or exchange data by navigating over such fields (i.e., by having particles that move following the gradient descent of a field, or by making them spread sort of data particles to be routed according to the shape of fields). The world in which such particles live and in which fields spread and diffuse can be either a simple (euclidean) metric world mapped in the physical space, or a network space mapped on the technological network, or it could also be a sort of relativistic world, in which shapes and distances in the environment are not "inherent" but are rather shaped by fields themselves (as in gravitational space-time).

Physical metaphors have been proposed to deal with several specific middleware-level aspects in dynamic network scenarios, such as in the Proto system (Beal and Bachrach, 2006) or in the TOTA middleware (Mamei and Zambonelli, 2005). Also, the physical inspiration is at the basis of the amorphous computing approach (Servat and Drogoul, 2002).

Physical metaphors appear suitable to facilitate coherent self-organized and self-adaptive behaviors even in large scale systems. Also, the conceptual tools available for engineering and controlling the spatial behavior and the dynamics of such systems are well-developed, most of them related to properly acting on the way fields propagate and dynamically change. Unfortunately, physical metaphors fall short in properly tolerating diversity and long-lasting adaptation. In fact, if one wants to preserve the simplicity of the eco-laws and of the supporting infrastructure (which is in charge of enforcing the eco-laws also by supporting field propagation and updates), one has to limit the amount of different fields in the system. However, to support very diverse components and behaviors (at a time and over time), the eco-laws must become complex enough to tolerate a wide range of different fields and propagation rules, with an overall increase in the conceptual complexity of the model and with an increase burden on the infrastructure, which has to support a large number of different laws for field diffusion and propagation.

3.2. Chemical metaphors

Chemical metaphors consider that the species of the ecosystem are sorts of computational atoms or molecules. Their properties can be described by some sort of semantic descriptions, as the computational counterpart of the description of the bonding properties of physical atoms and molecules, yet made dynamic by the fact that such description reflects the current state and the context of individuals. Accordingly, the laws that drive the overall behavior of the ecosystem are sort of chemical laws, that dictate how chemical reactions and bonding between components take place (i.e., relying on some forms of pattern matching between the semantic description of components), and that can lead to the production of aggregates (e.g., of aggregated distributed components), of new components (e.g., of composite components or of components spawned by existing ones) and to the growth/decay of a set of components. The world in which components live is typically formed by a set of localities, intended as the "solutions" in which chemical reactions can occur.

Chemical metaphors have been proposed to facilitate dynamic service composition in pervasive and distributed computing systems. Examples include chemically inspired discovery and composition services (Quitadamo et al., 2007), or chemically inspired coordination services (Viroli and Zambonelli, to appear).

In general, such chemical metaphors can effectively lead to local self-organizing structures (e.g., local composite services and local aggregates). Also, the same as real chemistry, chemical computational metaphors, can accommodate an incredible amount of different components and composites, yet with the same set of simple basic laws. In practice, this means that they can tolerate an increasingly diverse and evolving set of semantic description for components without affecting the basic eco-laws and without increasing the burden to the infrastructure. The key drawback of the chemical approach is that it typically relies on activities taking place within a locality, making it hard to naturally and easily enforce distributed self-organized behaviors. To this end, one should rely on complementing the chemical metaphor with physically or biologically inspired mechanisms to rule the diffusion of chemical substances across localities, and the creation of bonds between components in different localities (Viroli and Zambonelli, to appear). Clearly, this somewhat complicates the overall model and the supporting infrastructure.

3.3. Biological metaphors

Biological metaphors typically focus on biological systems at the small scale, i.e., at the scale of individual organisms (e.g., cells and their interactions) or of colonies of simple organisms (e.g., ant colonies). The species are, therefore, either simple cells or very simple animals, both of which can be seen as acting on the basis of very simple goal-oriented rules (e.g., move, find food, find mate). Similarly to physical systems, interactions take place by means of signals of various flavors (i.e., chemical pheromones) spread by components in the environment and to which components can react. However, unlike in physical systems, components are not necessarily passively subjects to pheromones around: the reaction of components to the sensed pheromones as well as their own

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

spreading of pheromones can depend on their current "mood" (which reflect its current state towards the achievement of a goal). Accordingly, within the biological metaphor, the laws of the ecosystem are only aimed at determining how such pheromones, depending on their specific flavors, should diffuse and/or evaporate, depending on their specific flavors. With this regard, we emphasize that pheromones, unlike fields, are typically more persistent than fields (i.e., they slowly evaporate and diffuse), which makes it easier for the infrastructure to support them, and which makes it also possible to exploit them as a sort of environmental memory, other than to reflect the current state of things in the environment. The world in which components live is typically a virtual computational landscape, either mapped on the network topology or on the physical space, that can influence the way pheromones diffuse and the way components can move over it.

Being somewhat similar to physical ones, biological metaphors too have been proposed to solve specific algorithmic problems in distributed network scenarios. For instance, to solve problems of synchronization or distributed resource allocation (Babaoglu et al., 2006) or to effectively coordinate the activities of distributed robots (Shen et al., 2004).

Biological metaphors appear very flexible in enabling the spatial formation of both localized and (by exploiting diffusion of pheromones and movements of individuals) distributed self-organizing and self-adaptive patterns of activity and computation. As in physical metaphors, the number of patterns that can be enforced by the spread of chemical pheromones and by the reactions of simple individuals seem quite limited, which does not match with the need to accommodate diversity and long-lasting evolution. Also in this case, the problems can be somehow circumvented by increasing the variety of pheromone flavors and characteristics. This solution, for biological metaphors, can be more easily supported by the infrastructure, since pheromones (unlike fields) rely on local diffusion and slow evaporation dynamics. An additional problem of biological metaphors is the inherent difficulty in understanding how to properly engineer and control the overall behavior of such systems. By thinking that, so far, the mechanisms of morphogenesis and biological self-organization in actual biological systems are not fully understood by scientists, one cannot hope to be able to properly enforce any desired form of control in complex biologically inspired systems.

3.4. Social and ecological metaphors

Social and ecological metaphors focus on biological systems at the level of animal species and of their finalized interactions, e.g., at the level of market-based (who needs to buy what from who, Cornforth et al., 2004) or trophical (who needs to eat who, Agha, 2008) interactions.

With reference to the case of trophical interactions, components of the ecosystem are sort of goal-oriented animals (i.e., agents) belonging to specific species (i.e., agent classes), that are in search of "food" resources to survive and prosper (e.g., specific resources or other components). The laws of the ecosystem determine how the resulting "web of food" should be realized, that is, they determines how and in which conditions animals are allowed to search food, eat, and possibly produce and reproduce, thus influencing and ruling the overall dynamics of the ecosystem and the interaction among individuals of different species. Similarly to chemical systems, the shape of the world is typically organized around a set of localities, i.e., of ecological niches (think at a set of local pervasive computing environments), yet enabling interactions and diffusion of species across niches.

In general, the adoption of such trophic metaphors promises to be very suitable for large-scale pervasive service ecosystems. In fact, other than supporting adaptive spatial forms of self-organization based on local economic or food-web forms of interactions, they also inherently support diversity (a large number of different agent classes can be involved in interactions) and long-lasting evolution (no matter who is engaged in interactions and how the individuals evolve over time, the overall laws of the ecosystems need not to be affected).

Economic and market-based models have been already extensively studied in recent years (Cornforth et al., 2004; Haque et al., 2005; Ramchurn et al., 2007) and they are well suited for large-scale Internet scenarios. However, they are possibly too complex and elaborated for pervasive computing scenarios, where simplicity of both individuals and interaction rules is a primary requirements (due to the presence of resource constrained computation and communication devices).

Based on the above consideration, on the fact that ecological models appears to be able to properly support spatiality, adaptivity, and long-lasting, and last but not least based on the fact that such models are largely unexplored (e.g., in Agha, 2008 the adoption of a trophic-based interaction model is argued but not put at work), we have decided to develop, explore and experiment with one of such models.

4. The ecological approach

In this section, we go into more details about the key characteristics of the proposed ecological (i.e., trophic-based) metaphor and about the modeling of the individuals according to it. Then, we introduce a case study to clarify the concepts expressed and to compare with alternative, more traditional approaches.

4.1. Key components

As already stated, ecological metaphors generally focus on biological systems at the level of animal species and of their interactions. In our specific trophic-oriented approach, the components of the ecosystem are sorts of goal-oriented animals (i.e., agents) belonging to a specific species (i.e., agent classes), that have the ego-centric goal of surviving by finding the appropriate food and resources.

In general, an ecological system can consider the presence of different classes of living forms (see Fig. 2). As far food-webs are concerned, we can consider the following key classes. Passive life forms (i.e., the flora system) do not actively look for food, although their existence and survival must be supported by nutrients that are in the space. Primary consumers (i.e., herbivores) need to eat vegetables to survive and prosper. Secondary consumers (i.e., carnivores) typically need to eat other animals to survive, though this does not exclude that can also act as primary consumers (eating vegetables too). The result of the metabolization of food by both primary and secondary consumers ends up in feeding lower-level "digesters" life forms (e.g., bacteria), densely spread in space, and that in their turn produce and diffuse resources and nutrients for the flora.

Let us now translate the above concepts in computational terms. Passive life forms represent the data sources and the computational/ memory/ communication resources of the ecosystem, which are not to be considered proactive and autonomous computational entities (i.e., they are not expected to act). Primary consumers represent those services that require to digest information to be of any use (i.e., to reach their goals), and that are computationally autonomous in pursuing such goals. Secondary consumers, instead, are those services that, to reach their goals, need the support of other computational services/agents, other than possibly of information sources. Digesters can be generally

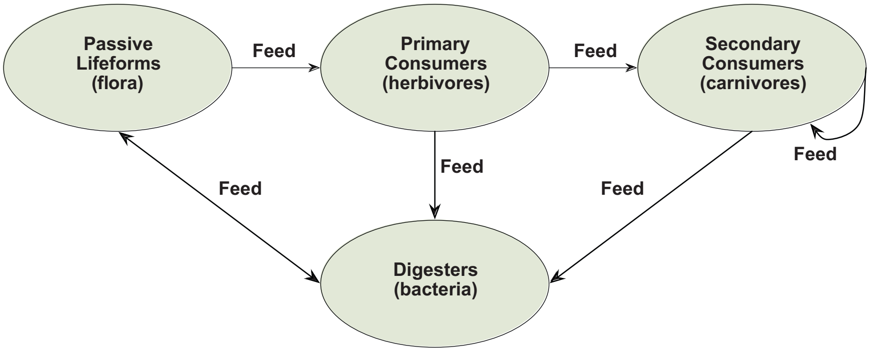

Fig. 2. Key elements of the food-web for an ecological system.

assimilated to all those background computational services that are devoted to monitor the overall activities of the system, and either produce new information about or influence the existing information and resources.

From Fig. 2, it is clear that the existence of such a food-web chain (or, in computational terms, of such inter-dependencies in the activities of agents) provides a close food-web loop. The existence of such loops is indeed a known basic regulatory ingredient to enable adaptation and self-organization in both natural and computational systems (Mamei et al., 2006). However, unlike in more structured approaches which consider the existence of a single (or of a limited set) of regulatory loops (Kephart and Chess, 2003), the ecological approach can make a number of related loops co-exist, depending on the number of different classes of individual and on their specific trophic relations.

As for the spatiality of the approach, the ecosystem space can be typically organized around a set of localities, i.e., of ecological niches (think at a set of local pervasive computing environments), yet enabling interactions and diffusion of species across niches. Each locality will determine how the different species organize to live, will describe how individuals of each species respond to the distribution of resources and other species, and how they alter these factors. Clearly, such way of organizing spatiality around open ecological niches provides for the existence of spatially distributed regulatory loops that ensure adaptation and self-organization at the global level, rather than at the level of individual niches.

Such ecological metaphor promises to be very suitable for local forms of spatial self-organization (think at equilibria in ecological niches), and are particularly suited for modeling and tolerating evolution over time (think at how biodiversity has increased over the course of evolution, without ever mining the health existence of life in each and every place on the Earth).

We are perfectly aware that understanding how to properly control the local and the global equilibria of real ecological systems is a difficult task, and so it would probably be very difficult also in their computational counterparts. Yet, and despite the challenges involved in it, we think that is an interesting and promising approach to explore.

4.2. Modeling of individuals

Let us now go into details about the modeling of the computational elements that populate the ecosystem (i.e., the ecosystem individuals) and of the mechanisms underlying their interactions (i.e., forming the basis of the eco-laws).

In our perspective, the key components of an individual include:

- A public semantic description of its own characteristics;
- A set of "needs" expressing which other individuals it needs to interact (i.e., "eat") with;
- A "happiness" status acting as a main drive for the individual activities;
- An internal logic for the maximization of its happiness, based on its needs and it happiness status, and relying on a set of allowed actions.

Active individuals typically express all the above components, passive ones (i.e., pure data items and resources) typically express only a public semantic description.

We recall that, in our model, each active individual needs food and resources to survive. That is, it needs other components to "eat" (i.e., to interact with), whether passive or active ones. These needs are specific for each specific individual/species. Generalizing upon dynamic models of semantic service matching (Quitadamo et al., 2007; Autili et al., 2009), the idea of our approach is that each individual expresses its own characteristics in a public semantic description. At the same time, each individual needs to eat some others, expresses its own needs in a sort of semantic "templates", describing the characteristics of the needed resources or individuals.

Then, based on the needs of an individual, and on the semantic descriptions of the individuals around it, a discovery process based on semantic matching takes place. Such process of matching typically occurs within a niche, though without excluding the possibility for one description to diffuse in spatially close niches and enabler cross-niches matching. The existence of a match of one individual with another needing it establishes a session of interaction between the two components, which we metaphorically abstract as an "eating" action by the individual that has found the right food. Pragmatically, we can think at a service component that has found the data/resources it needed or at a composite service that has found the needed computational partners required to fulfill its promises.

More formally, given the semantic description ind_a of an individual a and the needs $ind_b needs$ of another individual b, the process assumes the existence of a matching function:

$$match(ind_a, ind_b need),$$

that returns some value expressing the degree of match between individuals. The eco-laws of the ecosystem have the duty of automatically triggering possible matches in a niche (or across close niches) and of establishing a connection between matching individuals.

We do not go into details here on the specifics of such semantic representations and of the matching function. Other works exist detailing how such dynamic matching between components can occur and based on what kinds of representation (Quitadamo et al., 2007; Autili et al., 2009), and we forward the interested reader to them. What we want to emphasize here is that both the semantic descriptions of individuals and their expressed needs are dynamic entities that can change over time depending on the current state and the context of individuals. For instance, an individual that has

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

already fully satisfied some of its needs, will have to change its expressed needs to avoid keep on searching for something it does no longer need.

Upon occurrence of a match, the involved individuals can, according to their own specific behaviors, start interacting with each other (e.g., a primary consumer having found matching information, can decide to absorb or consume it). The specific capability (i.e., the set of actions allowed) within individuals will determine the course of such interactions. In general, though, the internal actions of an individual are mostly driven by the need to maximize its happiness.

The happiness value of an individual expresses the satisfaction degree of each individual, in other words it expresses how well an individual satisfies himself. The happiness value of an individual is generally a function of time, and it is affected by the previous happiness value and by the current state of the individual.

The happiness increases when the individual finds the proper resources to eat (i.e., a lot of matching individuals), because this implies it is effectively achieving the goals its has been conceived for. However, the happiness of an individual can also be affected by different situations in its niche (or in close niches), and specifically by the happiness of the other individuals in the niche.

Thus, we can distinguish between the instantaneous happiness of an individual, which is merely a function of the current matchings, i.e., for an individual i in a niche with n individuals:

$$H_i^{inst}(t) = f(match(ind_j, ind_i need),\ j = 1, \ldots, n) \qquad (1)$$

and the overall happiness of an individual, which is a function of the current happiness of the individual, of the previous happiness, and of the overall happiness of the other individuals in the niche, i.e., for an individual i:

$$H_i^{overall}(t) = f(H_i^{inst}(t), H_i^{overall}(t-1), H_j^{overall}(t-1),\ j = 1, \ldots, n). \qquad (2)$$

The set of allowed actions (usually aimed at egoistically increase the happiness of the individual) is the final elements that define an individual. Beside specific internal computational actions and beside interactions with other individuals, additional action capabilities of an individual can include reproducing itself, moving and/or replicating across niches, or even dying. For example, an individual that does not find suitable matches in the current niche can decide to move to another niche to search for matches that can increase its happiness.

As a final note, we emphasize that – as far as happiness and actions are concerned – similar proposals have been made in the area of goal-oriented and utility-oriented agent systems (Etienne de Sevin, 2005; Maes, 1991; Oliver Simonin, 2000). The peculiarity of our approach is to couple this with the overall natural inspiration applied to pervasive computing scenario and to exploit a flexible model of dynamic composition.

4.3. A case study

To clarify the proposed ecological approach, let us introduce a representative case study in the area of adaptive display systems, mostly inspired by the related work in this area by Ferscha et al., Ferscha and Vogl, Ferscha (2007), and by some previous experiences of our group in the area of adaptive advertisement (Ferdinando et al., 2009).

Consider a scenario with different kinds of pervasive devices spread on it, like a thematic park or an exhibition center, and in particular densely pervaded with digital wall-mounted screens where to display information for visitors, movies, advertisements, or whatever. We can consider each of these screens (i.e., the computational resources associated with each of them) as a spatially confined ecological niche. The overall goal of the system is to properly satisfy in a stable and balanced way all the stakeholders involved in it. These include: visitors looking for information, advertisers that want to display commercials on the screens, and the displays themselves, that have the goal of maximizing their own exploitation (and possible revenues from advertising companies).

Different classes of visitors will watch these screens to look for different information. Thus, we can think at sort of "user agents" executing on the users' PDAs that, once in the proximity of a screen (i.e., while finding themselves into that specific ecological niche) start looking for specific information to eat (i.e., to have it displayed). User agents would thus act as primary consumers with the goal to find the information required by users and maximizing users' happiness.

Concurrently, we can think at "advertising agents" that acting on behalf of some advertising company, roam from screen to screen in search of specific classes of user agents (i.e., those interested in specific types of information), with the ultimate goal of displaying advertisements where they could be more effective. Advertising agents would then act as secondary consumers.

Background monitoring agents, executing on each ecological niche and possibly interacting with each other, can contribute replicating and spreading information where it appears to be more appreciated, and can also contribute in supporting the spatial roaming of advertiser agents by directing them where they could find more satisfaction. Thus, they would act as digesters associated to each display.

Specifically, here we can conceive the existence of one "display agent" associated to each display and in charge of monitoring and ruling the display, also possibly making available to advertisers information about what is happening in close niches (thus supporting their movements). In addition, we can think at the presence of several "information agents" in charge of controlling the spreading and diffusion of information across displays. We emphasize that, although we conceptually prefer to think at information as something passive that is externally managed by sorts of digestor agents, for the sake of modeling it is also possible to think at information as something "active", that has its own "information agents" associated and with the goal of properly spreading information around.

The resulting patterns of interactions among agents can be defined accordingly to the food-web that we have already introduced in Fig. 2. This can be instantiated for the introduced case study as from Fig. 3. Information (or, which is the same, information agents) feed user agents, and these feed the advertiser agents. Display agents close the feedback loop in the system (between information agents and user agents; and user agents and advertiser agents) by affecting how the other agents move (and there included how information is spread by information agents). This is necessary for the ecosystem to overall self-regulate. It is expected that the system continuously evolve in time as a result of the interactions and actions occurring along such food-web.

As for the specific actions of agents, we can think at user agents moving from place to place and thus being in the view range of different displays (and thus looking specific information on them and being exposed to different advertisement), at advertising agents moving and replicating over the spatially distributed displays so as to capture the more suitable target audience, at display agents that provide to actuate the display, and at information agents that distribute and replicate information on need.

The feedback loop that derives from the above activities can contribute to properly rule the overall dynamics of the screen ecosystem, by continuously self-organizing and self-adapting the way information flows in the system, as well as the way advertising agents move, act, and coordinate with each other. The possibility of exerting control over the dynamics of the system is ensured by the possibility of injecting in the system additional classes of "digester agents" that can radically influence the dynamics of information

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

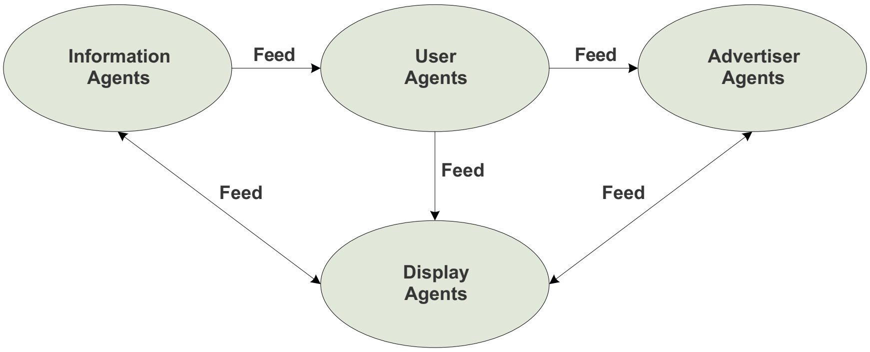

Fig. 3. Food-web for the ecosystem of displays case study.

diffusion and the activities of advertising agents. The adaptation of the system over time is ensured by the fact that it is mostly irrelevant, for the overall functioning of the system, what specific classes of information user agents want, or what the specific goal of advertising agent is. In fact, independently of the specific species of life forms that will populate the system, the basic eco-laws will ensure that such life forms will either find their way of living and their role in the system (e.g., as it can be the case of useful information and of advertising agents that find appropriate users to which to display their ads), or will simply disappear (as it can be the case of useless information or of advertisements no user is interested in).

Let us now detail the happiness functions for each class of the case study.

Users are happy when they see what they are interested in. The displays near to an user can satisfy (or not) his interests and consequently influence in his happiness in a positive or a negative way. In other words, if the users watch what they want/need in the displays around, they will feel happy. The function that gives us the instantaneous happiness of an user agent is defined by the following:

$$H^{inst}_{user}(t) = \frac{1}{n}\sum_{k=0}^{n} match(dT_k(t), user), \tag{3}$$

where $H^{inst}_{user}(t_0) = 0$.

n is defined as the number of displays that are around the user in a specific time t. The *match* function expresses the degree of satisfaction of the user when he watches the display k in a given time t. A match takes place when what it is displayed (dT—displayed task) it is in agreement with the interests of the user.

The overall happiness for one user agent is

$$H^{overall}_{user}(t) = H^{inst}_{user}(t) + \sum_{k=0}^{t-1} H^{overall}_{user}(k) - \Delta, \tag{4}$$

where $H^{overall}_{user}(t_0) = 0$.

The function is the result of the sum of the whole values that the overall happiness had ($\sum_{k=0}^{t-1} H^{overall}_{user}(k)$), and also the instantaneous/current happiness ($H^{inst}_{user}(t)$). Δ is a constant value that stand for the unhappiness degree of the user, it makes his happiness value decrease over time. This models the fact that the longer the user does not see something interesting, the more he gets unhappy. Eventually, if the user never sees something interesting, its happiness will lower to 0.

Advertiser agents are happy when they display their clips in an environment with many of happy users interested in their advertisements. The idea here is that a commercial clip is well received by users that are interested in it and that are "happy", i.e., that have not being bored by uninteresting clips so far. The function that evaluates the instantaneous happiness of advertiser agents is thus given by

$$H^{inst}_{adv}(t) = \frac{1}{n}\sum_{k=0}^{n} match(dT_k(t), adv)\left(\frac{1}{m}\sum_{k=0}^{m} H^{overall}_{user}(t-1)\right), \tag{5}$$

where $H^{inst}_{adv}(t_0) = 0$.

In this function n and m are defined as the number of displays and users, respectively, which are in the near environment of the advertiser agent in a specific moment t and then, influencing in his happiness value. This happiness is influenced by what the displays around show and by the average of the overall happiness of the users around ($1/m\sum_{k=0}^{m} H^{overall}_{user}(t-1)$). The happiness of users influences in the happiness of advertisers by the definition of the final goal of the last: to show their advertisements to happy and interested users. From this goal we can deduce that: what increases user happiness also increases advertiser happiness. By the other hand, the match function returns the degree of happiness when the advertiser agent watches the display k. A match takes place when what it is displayed (dT—displayed task) it is in agreement with the interests of the users around and of the advertiser agent, which can be: his own advertisement or an interested information for users. In a practical example, if the display k shows information (but this information keeps users happy), then the match function returns a positive value.

The overall happiness for one advertiser agent is given by

$$H^{overall}_{adv}(t) = H^{inst}_{adv}(t) + \sum_{k=0}^{t-1} H^{overall}_{adv}(k) - \Delta, \tag{6}$$

where $H^{overall}_{adv}(t_0) = 0$.

Similar to the overall happiness of users, this function includes the values of the overall happiness already happened ($\sum_{k=0}^{t-1} H^{overall}_{adv}(k)$) and the current/instantaneous one ($H_{adv}(t)$). Δ is the constant value that decreases the overall happiness of advertisers over time, expressing the degree of unhappiness.

The happiness of information agents is very simple to express. Information agent just want that the displays around them be interested in their information. The instantaneous happiness for an information agent is similar to the others instantaneous happiness described before and is given by

$$H^{inst}_{inf}(t) = \frac{1}{n}\sum_{k=0}^{n} match(dT_k(t), inf), \tag{7}$$

where $H^{inst}_{inf}(t_0) = 0$.

n is defined as the number of displays around the information agent, i.e., the number of displays that the information agent can see in a given instant t. The match function returns the degree of happiness of information agents when they see a display showing the task $dT_k(t)$ in a given time t.

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

The overall happiness for one information agent is obtained as the overall happiness described before, and is given by

$$H_{inf}^{overall}(t) = H_{inf}^{inst}(t) + \sum_{k=0}^{t-1} H_{inf}^{overall}(k) - \Delta, \tag{8}$$

where $H_{inf}^{overall}(t_0) = 0$.

The final goal of display agents is to show advertisements because they pay for advertise on it. Consequently, their happiness is proportional to the number of advertisements showed and the degree of advertisers' happiness, since displays want to keep their advertiser clients happy.

The function that evaluates the happiness of the displays is thus given by

$$H_{dis}^{inst}(t) = \frac{1}{n} \sum_{k=0}^{n} H_{adv}^{overall}(k), \tag{9}$$

where $H_{dis}^{inst}(t_0) = 0$.

n is defined as the number of advertisers that are around the display. The display's happiness is given by the average of the overall happiness of the advertiser agents that are in the niche.

The overall happiness for one display is given by

$$H_{dis}^{overall}(t) = H_{dis}^{inst}(t) + \sum_{k=0}^{t-1} H_{dis}^{overall}(k) - \Delta, \tag{10}$$

where $H_{dis}^{overall}(t_0) = 0$.

This function is defined in the same way that previous overall happiness.

4.4. Comparing with traditional SOA approaches

Following we describe how the presented case study could be architected according to the standard service-oriented architecture (SOA). The final result would be such a scramble of SOA, to rather suggest re-thinking from the scratch the architecture in ecological terms.

In general, SOA considers the inter-related activities of service components to be managed by various infrastructural (middleware) services such as: discovery services to help components get to know each other, context-services to help components situate their activities, orchestration services to coordinate interactions according to specific application logics, and shared data-space services to support data-mediated interactions. In this context, we have to set up a middleware server with the necessary infrastructural services to support the components of the considered scenario, i.e., displays, information, advertisers, users and personal devices.

In our scenario the interactions and the discovery of components are strongly bounded to spatial information (e.g., one display is interested only in the other near components—users, advertisers and information). In order to attempt this with SOA, we need to insert either sophisticated context-services to extract the necessary spatial information about components, or to embed spatial descriptions for each component into its discovery entry, inducing frequent and costly updates to keep up with mobility.

Furthermore, another requirement of our considered scenario is that its components be able to recognize relevant changes in their current environment and plan corrective actions in response to them. This will require notable communication and computational costs for all the components involved. Alternatively, or complementary, one could think at embedding adaptation logics into some specific server inside the middleware (e.g., in the form of autonomic control managers, Kephart and Chess, 2003). However, such logics would have to be very complex and heavyweight to ensure capability of adapting to any foreseeable situation, and especially hard for long-term adaptivity.

To reduce the identified complexities and costs and better match the characteristics of the scenario, we could think at a more distributed SOA solution, with a variety of middleware servers deployed in the infrastructure to serve, on a strictly local basis, only a limited portion of the overall infrastructure. For instance, we could install one middleware server for each of the available public displays. It will manage the local display and all local service components, thus simplifying local discovery and naturally enforcing spatial interactions. Adaptation to situations is made easier, thanks to the possibility of recognizing in a more confined way (and at reduced costs) local contingencies and events, and of acting locally upon them.

With the adoption of a distributed solution enforcing locality, the distinction between the logics and duties of the different infrastructural services fades: discovering local services and devices implies discovering something about the local spatial context; the dynamics of the local scenarios, as reflecting in the local discovery tables, makes it possible to have components indirectly influence each other (being their actions possibly dependent of such tables), as in a sort of shared data space model. This also induces specific orchestration patterns for components, based on the local logics upon which the middleware relies to distribute information and events among components and to put components in touch with each other.

To most extents, the concept of locality promoted by such distributed SOA solution makes the overall architecture of the system very similar to the organization in niches of our ecological approach. In addition, the fact that within a locality, the role of the various kinds of middleware services tend to vanish, again goes in the direction of making the SOA distributed solution similar to the ecological one. That is, a solution in which, within a locality, some local coordination rules (similar to eco-laws) support interactions.

In summary, it is of course possible to adopt a more traditional SOA solution, but its actual design would in the end force the adoption of solutions very similar to those that we propose, in a cleaner way, in our ecological approach.

5. Experimental results

In this section, we describe the simulation environment that we have realized to evaluate the effectiveness of the proposed approach in the introduced case study. Then, we present several results from the simulation experiments, assessing the approach in terms of dynamic behavior, stability and scalability.

5.1. The simulation environment

The simulation environment has been realized with the Recursive Porous Agent Simulation Toolkit (Repast, repast.sourceforge.net).

As from Section 4.3, the simulated scenario of the case study considers a 2D space (i.e., the area of a thematic park) populated with displays. Therefore, a number of spatially situated agents belonging to different classes (users, advertisers, display, and information agents) act, interact and evolve according to a simple set of eco-laws (driving their matchings and interactions) and each with the ego-centric goal of maximizing their individual happiness.

In the simulation environment, displays (and thus display agents over them) have a fixed position and each represent a niche. Displays are assumed to be able to display both information (some data they own and that can be useful to users) and advertisements (based on the actions of advertiser agents). Display agents have indeed the goal of showing information and advertisements, and their happiness is proportional to the overall happiness of advertiser agents and user agents that live in their niche, as from previous section.

Users (i.e., user agents) randomly move in the simulation space, that is, be moving close to specific displays, they move from niche to niche. Each user agent is assumed to have an individual profile, expressing its preferences in terms of information and advertisements he would like to see. According to these preferences a matching function is defined for the instantaneous happiness of a user watching a display, that measures the degree of satisfaction of the user when watching information and/or advertisements. The matching function returns a value between 0 and 1, depending on the degree of match between its preferences and the information/advertisements currently being shown in the display in which he is situated. In any case, we emphasize that in the simulation user agents do proactively pursue the goal of increasing their happiness, but are simply exposed to information and advertisements around.

Advertiser agents that reside on niches have the goal – that they proactively pursue – of showing a specific advertisement to a well-targeted audience. Again, a matching function varying between 0 and 1 is defined. The matching affects the happiness of advertisers depending on how well the overall audience (i.e., the users currently in that display niche) matches the advertisements. Clearly, the currently displayed advertisement is selected as the one best matching the current audience. Advertiser agents can move to a different niche in the case no suitable matching is found in the current one.

Information agents reside on niches and manage specific types of information. That is, by monitoring the overall situation on one niche, can affect the information stored (and displayed) on that niche. That is, by increasing in one niche that information that results being of use to the classes of users in that niche, and by deleting useless information instead. Also information agents can move from niche to niche to properly distribute on need the information they hold.

Figure 4 shows a graphical screenshot of the simulation scenario with 100 advertiser agents, 500 user agents, 100 information agents, 25 displays and 25 niches. The squares represent the displays and the little bodies represent the users. Displays can show different colors that represent what they are showing: information, advertisements, both of them, or simply nothing. To avoid overcrowding, the figure does not show advertiser and information agents that reside on the displays.

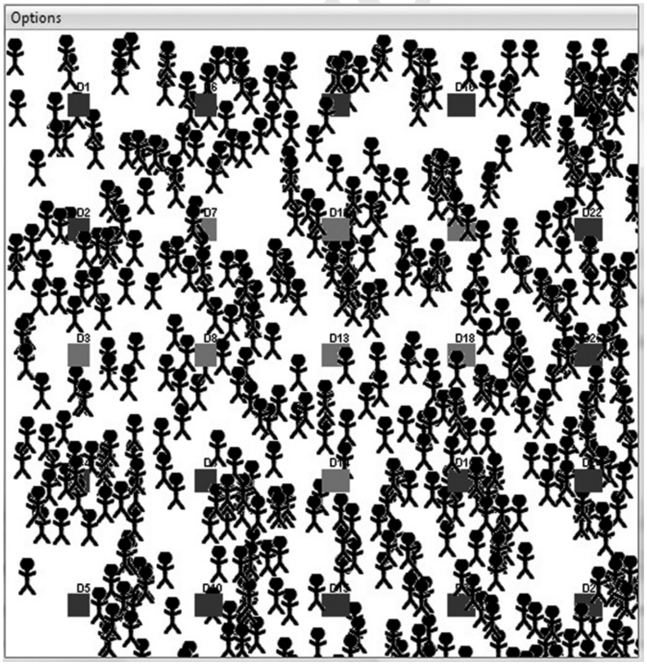

Fig. 4. Screenshot of simulation with 500 users and 25 displays.

Despite some limitations of the current simulation environment (that is, the number of agents is static since agents cannot die or reproduce; information and advertiser agents are not capable of dynamically adapting their behavior and interacting with each other, but rather follow a set of individual static rules), experimenting over it enables to get useful insights on the overall behavior of our proposed ecosystem approach.

5.2. Overall ecosystem behavior

In the following we use two different simulation scenarios, which we call for simplicity *SS*1 and *SS*2, adopting the following parameters:

- *SS*1: 25 niches, 500 user agents, 100 advertiser agents, 100 information agents;
- *SS*2: 25 niches, 100 user agents, 100 advertiser agents, 100 information agents.

Let us first analyze the overall behavior of the ecosystem in terms of how the average happiness of the agents involved evolve over time.

Figures 5(a) and (b) show the evolution of happiness for scenarios *SS*1 and *SS*2, respectively. The figures show that, after a short transitory period, the overall average happiness stabilizes to a nearly steady value for all the agents involved. That is, the ecosystem reaches a sort of global equilibrium in which its global parameters are stable, independently of the specific parameters of the simulation. Indeed, such stability is reached not only in scenarios *SS*1 and *SS*2, but also in other scenarios with different number of agents and niches involved.

In any case, such apparent equilibrium is by no means the expression of a static situation. Rather, as the simulation continues executing and the agents in it continue acting and moving, it is an expression of the capability of the ecosystem as a whole to globally reach in a self-organizing way a coherent dynamic behavior.

To better analyze these results, let us see what happens at the level of individual niches, where we will discover many complex behavior that are hidden behind the above discussed global stability.

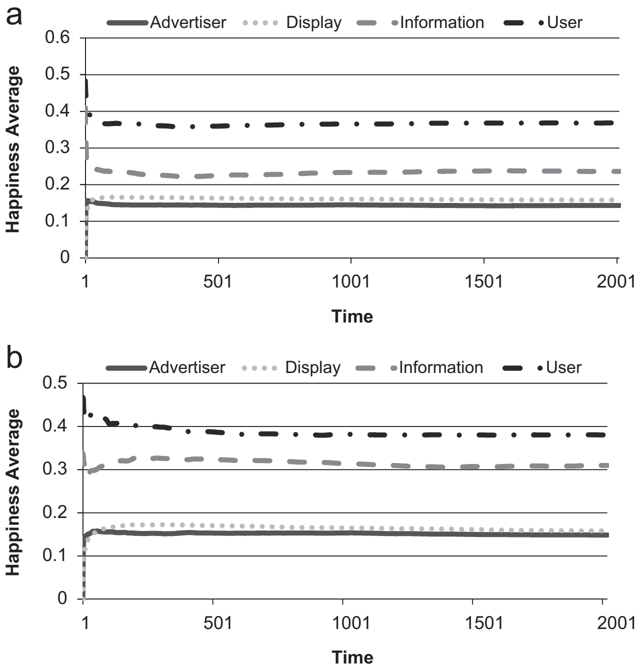

Fig. 5. Average happiness of agents for the whole ecosystem. (a) *SS*1 and (b) *SS*2.

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

5.3. Niche-level behavior

Let us see how things evolve in a single niche over time. Figs. 6 and 7 show the evolution of the happiness and of the number of agents, respectively, in one of the 25 niches of *SS*1 (niche 11). However, the patterns of evolution are very similar for all niches.

What we see here is that the stable balance already discussed at the global level is achieved despite a high-degree of dynamics inside a niche, where the average happiness of the various classes of agents tends to vary over time, along with the number of agents themselves.

By comparing then Figs. 7 and 8 (the latter referring to *SS*2), one can see that the variability in the number of agents in one niche is much higher when there are a comparable number of user and advertiser agents. In this case, in fact, advertiser agents have to be much more mobile to find proper places where to effectively show their advertisements, because the diversity of the situation of the various niches can be instantaneously more various. However, as from Figs. 5(a) and (b), this does not affect the capability of the system of reaching a global stable equilibrium.

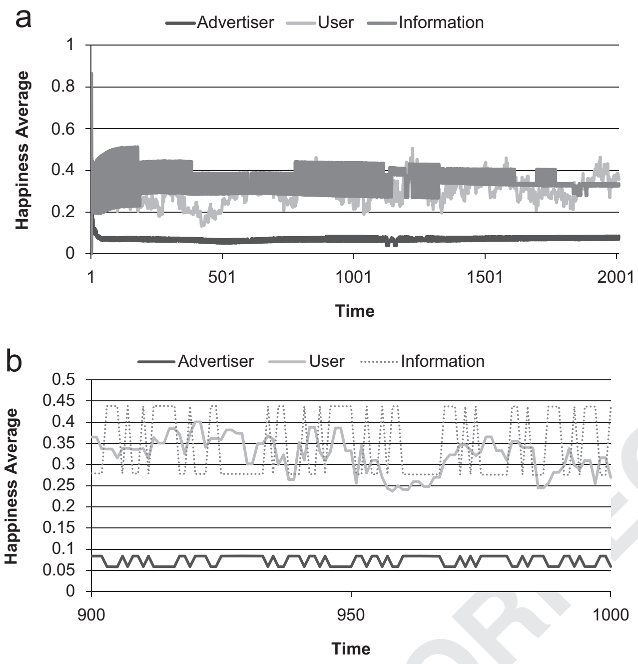

Fig. 6. Happiness of agents over time in one niche (*SS*1). The figure (b) is a zoom view of (a).

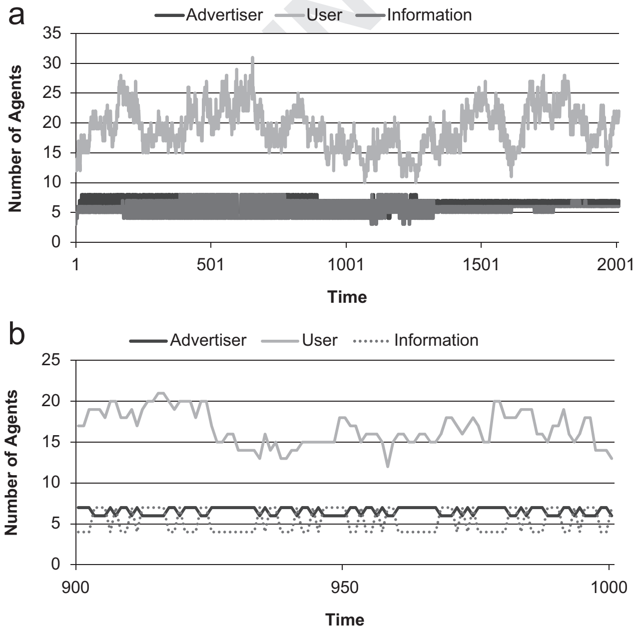

Fig. 7. Number of agents over time in one niche (*SS*1). The figure (b) is a zoom view of (a).

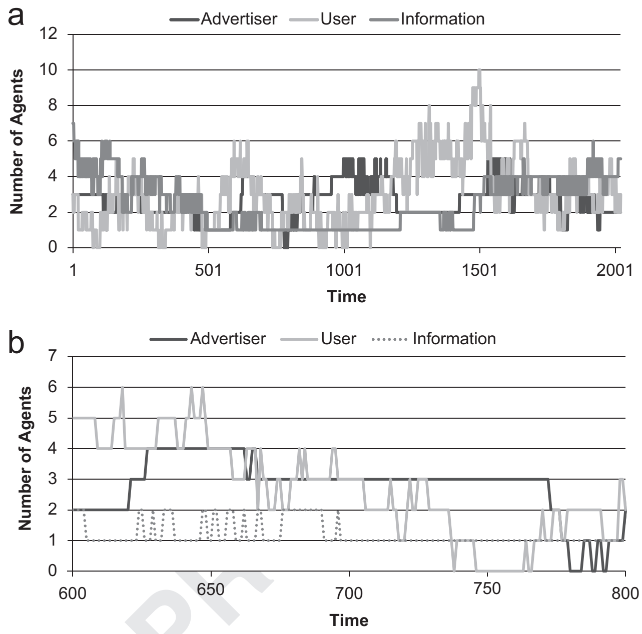

Fig. 8. Number of agents over time in one niche (*SS*2). The figure (b) is a zoom view of (a).

Figures 9(a) and (b) show the happiness average and the number of agents average for each class of agent, for each niche of the *SS*1 simulation scenario.

By looking at Fig. 9(a), we can see that the happiness average of user agents is well balanced in each of the niches. That is, the specific dynamics of the ecosystem, as enforced by the happiness functions of its various agents and by their interaction, have the effect of trying to make users as happy as possible, which results in users being nearly equally happy in every niche. Such balance of user happiness goes at the price of somehow unbalancing the happiness of the agents of other classes.

Such effect also is reflected in Fig. 9(b), where it can be seen that the number of advertiser and information agents notably varies from niche to niche, to due to the fact that they are forced to be mobile in order to satisfy their own happiness.

It is also interesting to refer to Figs. 9(c) and (d), showing the number of the displayed task average and requested task average, respectively, for each niche (again for *SS*1).

In particular, one can see that there is an overall balance across niches between the amount of displayed information and of displayed advertisement (Fig. 9(c)), which was a necessary condition to make users overall happy. However, this is achieved at the price of some relevant unbalance in the expectations of advertisers and information agents, which requested a number of task much more differentiated across niches.

5.4. Scalability

To evaluate the capability of the ecosystem to preserve its behavior in large scenarios, we have evaluated it under scenarios with a larger number of users and a larger number of niches.

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

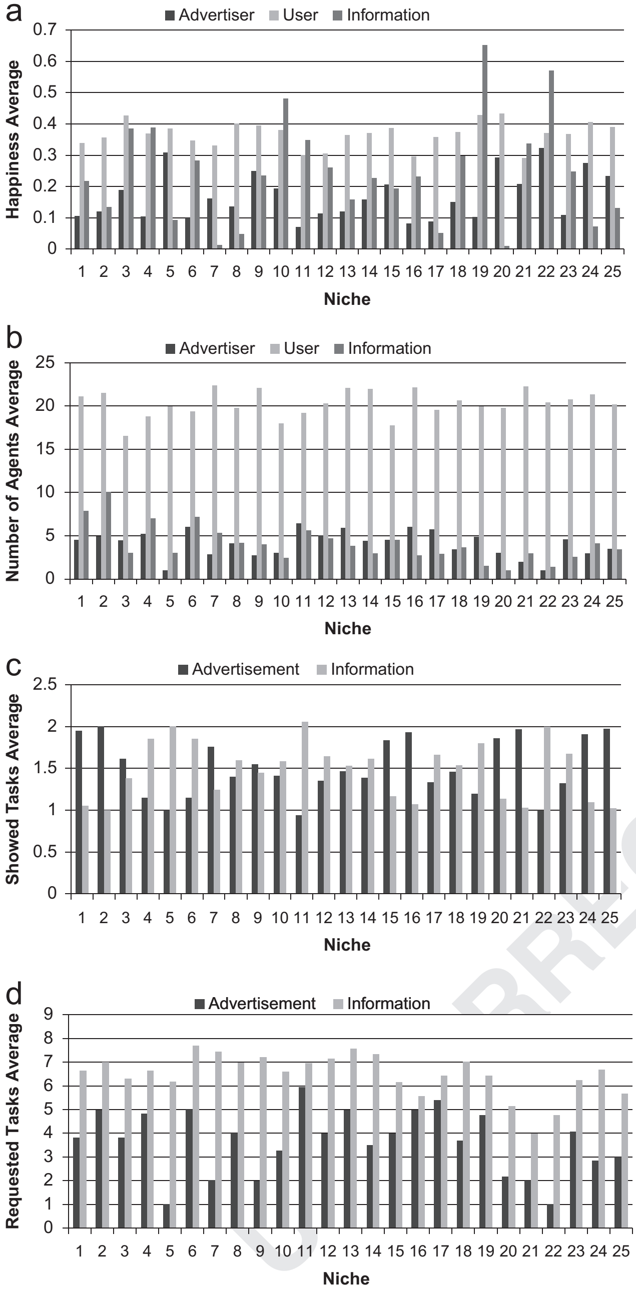

Fig. 9. Happiness of agents (a), number of agents (b), number of showed tasks (c) and number of requested tasks (d) average by niches (*SS*1).

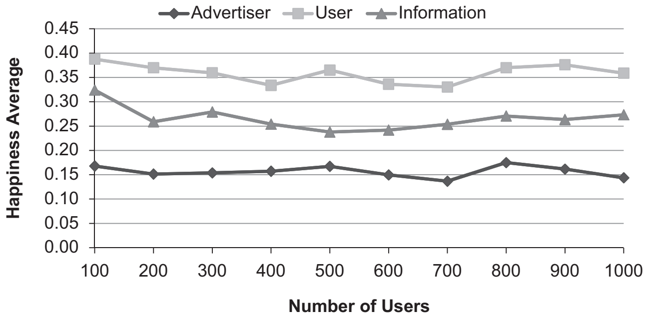

Fig. 10. Happiness of agents average with different number of users (100 information agents, 100 advertiser agents and 25 niches).

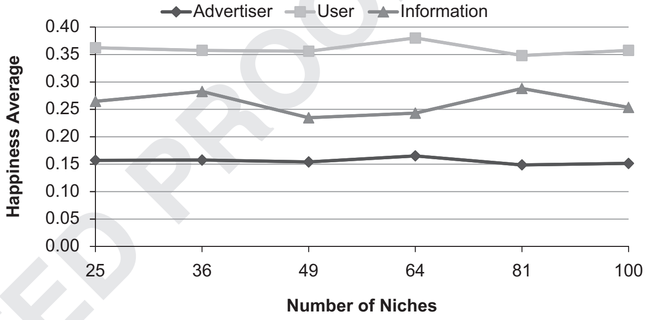

Fig. 11. Happiness of agents average with different number of niches (it begins with *SS*1 and then increases proportionally).

Figure 10 shows the average happiness of agents when increasing the number of users involved. Interestingly, one can see that there are no substantial changes in the happiness of the various agent classes involved in the simulation. In other words, the overall behavior of the system keeps stable even by changing the number of users, showing that its overall properties are maintained independently of the actual number of agents involved in the scenario.

Figure 11, on the other hand, shows the average happiness of agents when increasing the number of niches (while preserving the same average number of user agents). Specifically we consider 25, 36, 49, 64, 81 and 100 niches with 500, 720, 980, 1280, 1620 and 2000 of user agents, respectively. Also in this case, no substantial changes can be perceived as the number of niches (and, thus, the size of the system) increases. This prove that our ecosystem approach, being based on local interactions only, is capable of preserving its properties even in the large.

In summary, the above two figures testify that the proposed ecological approach is suitable for large-scale systems with a massive number of agents.

5.5. *Influence of other parameters*

To further assess the robustness of our approach, we have evaluated how its behavior changes by tuning both the parameters involved in the matching functions and by varying the number of tasks that a display can concurrently perform (i.e., the number of information and advertisements it can concurrently show).

Varying the parameters of the matching functions implies modifying the rate of happiness increase (or decrease) in agents. For example, increasing the average return value of the matching between users and advertisements implies making users more tolerant to see advertisement.

Figure 12 shows exactly the effect of increasing the average return value of the matching function between users and advertisement. If users are more tolerant to see advertisements, which means that the match function when users see advertisements will return a higher value, then users and advertisers happiness increase, since the system shows more advertisements because. Information happiness decreases because the times to show its information also decrease (Fig. 12).

From the definition of the happiness of advertisers, we know that if in a niche the display shows interesting information that increases the happiness of users, this will also increase the happiness of advertisers according to their degree of tolerance on seeing information. In Fig. 13, we increase the degree of happiness when advertisers watch information. In other words, advertisers are more tolerant to see information, which means that

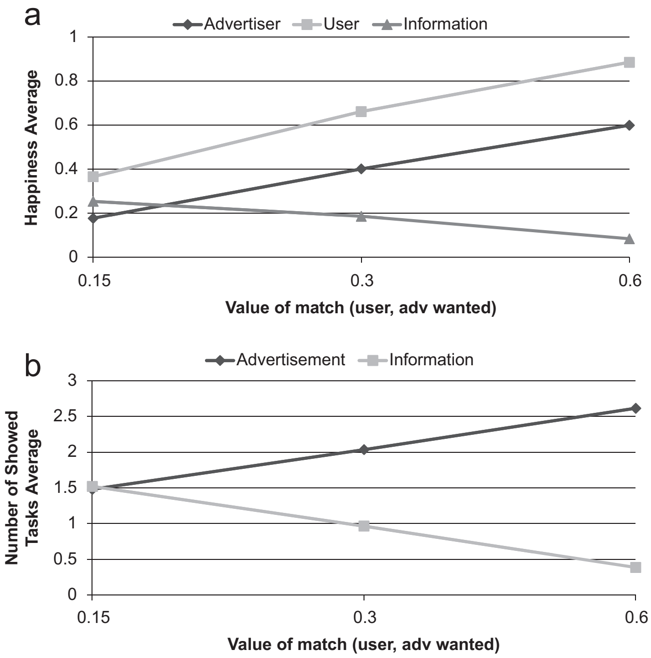

Fig. 12. Happiness of agents (a) and number of showed tasks (b) average by varying the degree of happiness of users when they see advertisements (*SS*1).

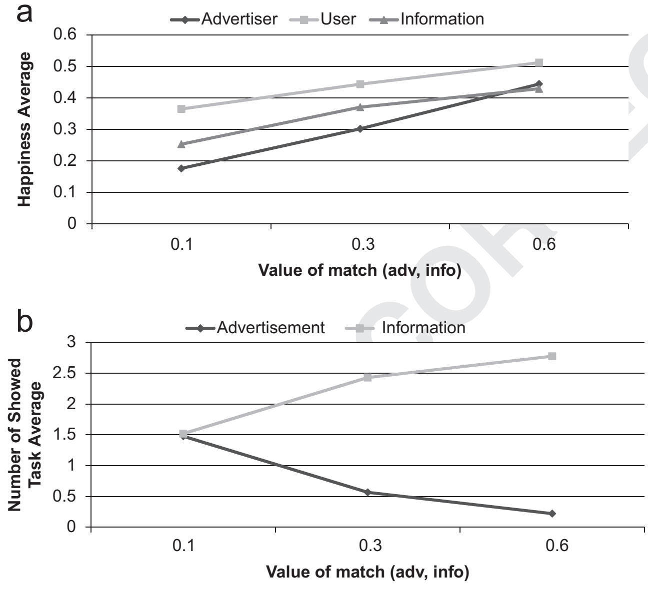

Fig. 13. Happiness of agents (a) and number of showed tasks (b) average by varying the degree of happiness of advertisers when they see information (*SS*1).

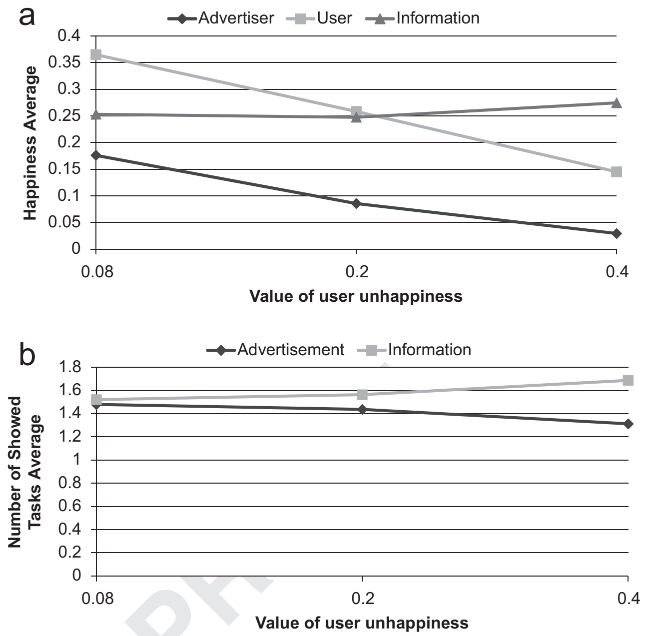

Fig. 14. Happiness of agents (a) and number of showed tasks (b) average by varying the degree of unhappiness of users (*SS*1).

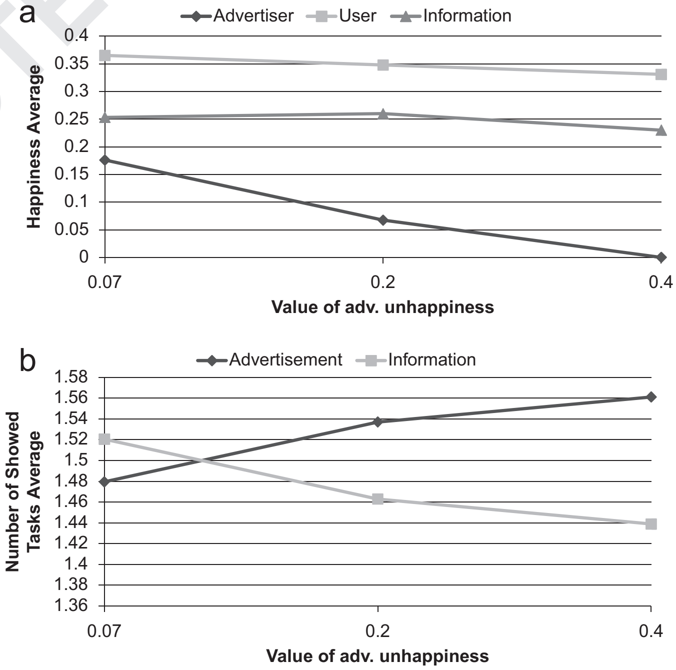

Fig. 15. Happiness of agents (a) and number of showed tasks (b) average by varying the degree of unhappiness of advertisers (*SS*1).

the match function between an advertiser and a displayed information will return a higher value.

As expected, the happiness of agents increases, also the happiness of advertisers (since their level of tolerance also increases), even though, the number of advertisers showed decreases and the number of information showed increases.

The other parameter that influences in the overall happiness functions is the value of the unhappiness constant (see the descriptions of the overall happiness functions in Section 4.3). If we increase the degree of unhappiness of users (Fig. 14), that means we have users less tolerant, the system shows more information looking to increase the happiness of users and consequently the happiness of advertisers. On the other hand, if we increase the degree of unhappiness of advertisers (then we have advertisers less tolerant), the system tries to balance between the number of information and of advertisements showed, in order to increase the happiness of the (intolerant) advertisers. (see Fig. 15). In both situations, the system tries to stabilize the happiness of their components, by playing with the number of times that each display shows information and/or advertisements.

As a final test, we modified the simulation by varying the number of tasks (information and/or advertisement) that each display can concurrently show. Fig. 16(a) shows how the average happiness of agents is proportional to the number of tasks that each

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

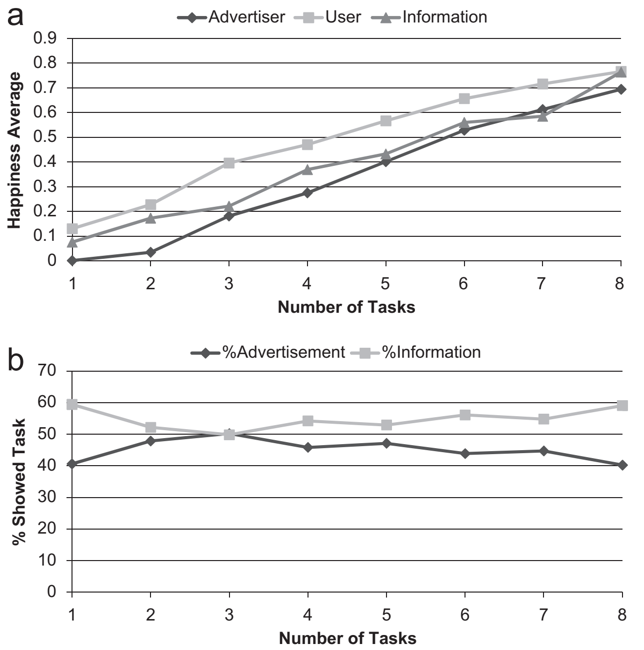

Fig. 16. Happiness average (a), and percentage of information and advertisements showed (b) by varying the number of task that each display can show simultaneously (*SS*1).

display can show simultaneously. This is simply due the fact that each display can accommodate at a time the needs of a larger number of users and advertisers. However, this does not change the overall behavior of the ecosystem. For instance, Fig. 16(b) shows that the relative percentage of information and advertisements shown on the displays does not substantially change. As more information and advertisements can be concurrently shown, the percentage of information tends to grow over advertisement. However, the overall balance between advertisement and information is preserved, due to the inherent dynamics of the regulatory feedback loops which are part of the system.

5.6. Summary

The above presented simulation experiments, although performed within a single case study and still exhibiting some limitations, shows very interesting properties. In particular,

- In the proposed ecological approach, the components of the system are able to reach stable self-organized global system behaviors, despite the inherent dynamics of the system itself;
- The approach promises to be scalable to very large-scale systems with a massive number of agents;
- The approach appears to exhibit very little sensitivity to variations in the parameters of agents and ecosystems.

6. Other application scenarios

In this section, we sketch other scenarios where our ecological concepts can be applied. The goal is to show that our proposed approach is a general-purpose one.

Consider first the general scenario of *smart buildings* (e.g., smart offices, smart houses, or smart rooms). The overall goal here is to properly adapt in a stable and balanced way the overall functioning of the building infrastructure (e.g., intensity of light, temperature, music, etc.). For sake of simplicity, we do not focus here on other potential issues for smart buildings, such as safety or activity support to humans.

The components involved in the overall smart building management are many. For each of the rooms and corridors of a building, we can consider the presence of sensors (e.g., cameras, thermal sensors, noise level sensors, etc.) and actuators (e.g., heaters, air conditioners, motorized curtains, music speakers, etc.). Then, we have human users, living, working, and moving in the building. We can assumed they have some sort of portable device (e.g., a PDA or a smart phone), via which their specific preferences (e.g., the desired heating level) can be expressed or sensed.

Clearly, the overall functioning of the building and of its temperature, light, or sound actuators, has to account for a number of possibly contrasting objectives. If there is only one user in a room, the status of such room (i.e., its heating, light, or music levels) can fit his/her preferences, thus keeping he/she comfortable and “happy”. However, when several users with different needs are in a room, some compromise status should be found. Usually, in this case, there are contradictory preferences of users, for which a mediation point is needed, e.g., one user likes the temperature at 30 °C and other at 20 °C, then the temperature of the room will be set at 25 °C. Furthermore, since users can come and go from a room, the status of the room must be continuously updated to the current situation (i.e., to the preferences of the users currently in that room).

In addition to the local goals to be achieved within each room and corridors, a proper functioning of the overall system, has to account for some more global factor, such as the global energy consumption (as induced by heating and air conditioners, as well as by lightning plants). Thus, the dynamics within each room and corridor that meet the specific needs of users, has to co-exist with the need for more global stable behaviors as far as the global parameters of energy consumption are concerned.

By modeling the overall application in terms of our ecological approach, and by exploiting its self-organizing features, the above needs of adaptation can be effectively achieved.

The ecological model of this application scenario, will consider: each room of the building or house as an ecological niche; all the data coming from the room sensors as well as data expressing users' preferences will be passive life forms; software components in each niche, acting as sorts of herbivors, can be in charge of evaluating the current state of a niche (by absorbing sensorial data) and producing planning decisions for the actuators; the actuators act as sort of carnivors that digest the decisions of the planners and act accordingly.

Putting such system at work in a dynamic scenario, will lead to a behavior very similar to that already observed for the pervasive display scenario (Cfr. Figs. 5 and 6). In spite of a high dynamic at the level of each room/niche, a globally stable behavior is exhibited at the global level. That is, the self-organizing activities of the ecosystem spontaneously find the proper trade-off between local and global goals.

A second exemplary scenario where our approach can effectively applied is *dynamic traffic management*. The overall goal here is to adaptively control the traffic in an urban area, avoiding crushes and traffic jams, satisfying at the best the needs of citizens and drivers, and overall keep pollution and noise at reasonable levels.

Also in this scenario many different devices and actors are involved: adaptive traffic lights, cars, drivers, pedestrian, cyclist, diverse traffic sensors, GPS devices, etc. The same as in the smart buildings scenarios, the human actors may have preferences and constraints, as far as their driving route and their schedule are concerned. For instance, there may be a driver that wants to reach home by avoiding tolls and has no hurry, while another may prefer the fastest route whatever it costs. Thus, there is again the need of conciliating the possibly contrasting needs of users driving in specific portions of the city with the global goals of balancing the traffic in different urban areas and, at the same time, reducing the overall pollution and whatever further factor may impact the quality of life.

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003

Modeling this application scenario in ecological terms implies: associating niches to crossings and streets; assimilating sensors and user profiles to passive life forms; considering drivers and cars as sorts of herbivors that consume data and resources to egoistically pursue their own goals; considering the existence of further agents (e.g., associated to an adaptive street light) that act as secondary consumers and interact with driver agents to understand their planned activities and influence them according to local goals (e.g., avoid crashes and traffic jams in a crossing); finally, considering the presence of monitoring agents that exchange information with each other to plan for the achievement of global goals in dependence of the local activities.

At the level of local and global behavior of the resulting urban traffic ecosystem, we can again observe the co-existence of local dynamics and globally stable behaviors.

7. Conclusions

In this paper, we have elaborated on the idea of modeling and developing the next generation of pervasive service framework by getting inspiration from ecological systems. That is, by conceiving future pervasive service frameworks as a spatial ecosystem in which services, data items, and resources are all modeled as autonomous individuals (agents) that locally act and interact according to a simple set of well-defined “eco-laws”.

We attempted to clarify these ideas through representative case study scenarios and to assess the potential of our approach via an extensive set of simulation experiments on a specific case study. Beside the current limitations of our simulation environment, the presented results suggest that the proposed ecological approach has the potential to act as an effective general-purpose framework for spatially situated and adaptive pervasive service ecosystems.

Beside the promising results, there are still a lot of open issues to exploit to turn our idea into a practically usable one. First, more simulations on a larger set of case studies are needed. Second, the proposed modeling for individuals and interactions needs to be better formalized and its properties more formally analyzed. Third, the security threats of our approach have to be identified. Finally, of course, there is need to put the approach at work in a real implementation and to test it on the field.

Acknowledgment

The work supported by the SAPERE project (EU FP7-FET, Contract no. 256873).

References

Agha G. Computing in pervasive cyberspace. Communications of the ACM 2008;51:68–70.

Autili M, Benedetto P, Inverardi P. Context-aware adaptive services: the plastic approach. In: FASE ’09: proceedings of the 12th international conference on fundamental approaches to software engineering. Berlin, Heidelberg: Springer-Verlag; 2009. p. 124–39.

Babaoglu O, Canright G, Deutsch A, Caro GAD, Ducatelle F, Gambardella LM, et al. Design patterns from biology for distributed computing. ACM Transactions on Autonomous and Adaptive Systems 2006;1:26–66.

Beal J, Bachrach J. Infrastructure for engineered emergence on sensor/actuator networks. IEEE Intelligent Systems 2006;21:10–9.

Brazier FM, Kephart JO, Parunak HVD, Huhns MN. Agents and service-oriented computing for autonomic computing: a research agenda. IEEE Internet Computing 2009;13:82–7.

Castelli G, Rosi A, Mamei M, Zambonelli F. A simple model and infrastructure for context-aware browsing of the world. In: PERCOM ’07: proceedings of the fifth IEEE international conference on pervasive computing and communications. Washington, DC, USA: IEEE Computer Society; 2007. p. 229–38.

Cornforth D, Kirley M, Bossomaier T. Agent heterogeneity and coalition formation: investigating market-based cooperative problem solving. In: Proceedings of the third international joint conference on autonomous agents and multiagent systems. Washington, DC, USA: IEEE Computer Society; 2004. p. 556–63.

Dobson S, Denazis S, Fernández A, Gaïti D, Gelenbe E, Massacci F, et al. A survey of autonomic communications. ACM Transactions on Autonomous and Adaptive Systems 2006;1:223–59.

Etienne de Sevin DT. An affective model of action selection for virtual humans. In: AISB ’05: proceedings of agents that want and like: motivational and emotional roots of cognition and action symposium at the artificial intelligence and social behaviors, 2005.

Estrin D, Culler D, Pister K, Sukhatme G. Connecting the physical world with pervasive networks. IEEE Pervasive Computing 2002;1:59–69.

Ferdinando AD, Rosi A, Lent R, Manzalini A, Zambonelli F. Myads: a system for adaptive pervasive advertisements. Pervasive and Mobile Computing 2009;5: 385–401.

Ferscha A. Informative art display metaphors. In: Fourth international conference on universal access in human–computer interaction, Lecture notes in computer science, vol. 4555. Springer; 2007. p. 82–92.

Ferscha A, Riener A, Hechinger M, Schmitzberger H. Building pervasive display landscapes with stick-on interfaces. In: Workshop “Information Visualization and Interaction Techniques”, associated with CHI’06 international conference. Quebec, Canada, 2006. p. 9.

Ferscha A, Vogl S. The webwall. In: Proceedings of the Ubicomp 2002 workshop on collaboration with interactive walls and tables. Göteborg, Sweden, 2002.

Haque N, Jennings NR, Moreau L. Resource allocation in communication networks using market-based agents. Knowledge-Based Systems 2005;18:163–70.

Herold S, Klus H, Niebuhr D, Rausch A. Engineering of it ecosystems: design of ultra-large-scale software-intensive systems. In: ULSSIS ’08: Proceedings of the second international workshop on ultra-large-scale software-intensive systems. New York, NY, USA: ACM; 2008. p. 49–52.

Huhns MN, Singh MP. Service-oriented computing: key concepts and principles. IEEE Internet Computing 2005;9:75–81.

Kephart JO, Chess DM. The vision of autonomic computing. Computer 2003;36: 41–50.

Locatelli MP, Loregian M, Vizzari G. Artificial societies in a community-based approach to ambient intelligence. The Computer Journal 2010;53:1152–68.

Maes P. The agent network architecture (ana). SIGART Bulletin 1991;2:115–20.

Mamei M, Zambonelli F. Field-based coordination for pervasive multiagent systems (springer series on agent technology). Secaucus, NJ, USA: Springer-Verlag, New York Inc.; 2005.

Mamei M, Menezes R, Tolksdorf R, Zambonelli F. Case studies for self-organization in computer science. Journal of Systems Architecture 2006;52:443–60.

Oliver Simonin JF. Modeling self satisfaction and altruism to handle action selection and reactive cooperation. In: SAB ’00: simulation of adaptive behaviors, 2000. p. 314–23.

Quitadamo R, Zambonelli F, Cabri G. The service ecosystem: dynamic self-aggregation of pervasive communication services. In: SEPCASE ’07: proceedings of the first international workshop on software engineering for pervasive computing applications, systems, and environments. Washington, DC, USA: IEEE Computer Society; 2007. p. 1.

Ramchurn SD, Sierra C, Godo L, Jennings NR. Negotiating using rewards. Artificial Intelligence 2007;171:805–37.

Roy PV, Haridi S, Reinefeld A, Stefany J-B, Yap R, Coupaye T. Self-management for large-scale distributed systems: an overview of the selfman project. In: Formal methods for components and objects. Lecture notes in computer science, vol. 5382. Springer-Verlag; 2008. p. 153–78.

Servat D, Drogoul A. Combining amorphous computing and reactive agent-based systems: a paradigm for pervasive intelligence? In: AAMAS ’02: proceedings of the first international joint conference on autonomous agents and multiagent systems. New York, NY, USA: ACM; 2002. p. 441–8.

Shen W-M, Will PM, Galstyan A, Chuong C-M. Hormone-inspired self-organization and distributed control of robotic swarms. Autonomous Robots 2004;17: 93–105.

Ulieru M, Grobbelaar S. Engineering industrial ecosystems in a networked world. In: Fifth IEEE international conference on industrial informatics. IEEE Press; 2007. p. 1–7.

Viroli M, Zambonelli F. A biochemical approach to adaptive service ecosystems. Information Sciences 180; to appear. Q1

Want R. An introduction to RFID technology. IEEE Pervasive Computing 2006;5: 25–33.

Zambonelli F, Mamei M. Spatial computing: an emerging paradigm for autonomic computing and communication. In: First international workshop on autonomic communications. Lecture notes in computer science, vol. 3457. Springer; 2005. p. 44–57.

Zambonelli F, Viroli M. Architecture and metaphors for eternally adaptive service ecosystems. In: Intelligent distributed computing, systems and applications, proceedings of the second international symposium on intelligent distributed computing, Studies in computational intelligence, vol. 162. Springer; 2008. p. 23–32.

Please cite this article as: Villalba C, Zambonelli F. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. J Network Comput Appl (2010), doi:10.1016/j.jnca.2010.12.003