Mechanisms of Self-Organization in Pervasive Computing

Nicola Bicocchi, Marco Mamei, Franco Zambonelli

Abstract—**The mass deployment of sensors and pervasive computing systems expected in the next few years, will require novel approaches to program and gather information from such systems. Suitable approaches will be general purpose, independent of a specific scenario and sensor deployment, and able to adapt autonomically to different scales and to a number of unforeseen circumstances. This paper focuses on the requirements and issues of upcoming pervasive computing scenario, and surveys current research initiatives to deal with them. In particular researches addressing data retrieval and aggregation, macro-programming, and data integration in pervasive computing infrastructures will be detailed. Overall, the paper illustrates our ideas on collecting information from both sensor systems and Web resources and on linking them together in overlay knowledge network offering applications comprehensive and understandable information about their computational environment.**

Index Terms—**Pervasive computing, Sensor network, RFID tags, Self-organization.**

I. INTRODUCTION

In the near future, computer-based systems will be embedded in all our everyday objects and in our everyday environments. These systems will be typically communication enabled, and capable of coordinating with each other in the context of complex mobile distributed applications.

Current realizations of such scenarios, mainly in research labs, focus on special purpose systems, tailored for a specific application task. This specialization comes rather directly from the extremely limited capabilities of pervasive devices, that impose to rule out ancillary and general properties for the sake of optimization. In sensor network scenarios, for example, in order to be compliant with the thin battery budget of each sensor, applications rely on special purpose algorithms tailored for a specific sensors' deployment and for a specific set of data to be measured [WerL06].

Nicola Bicocchi is with the Dipartimento di Scienze e Metodi dell'Ingegneria. Università di Modena e Reggio Emilia, Italy; e-mail: mamei.marco@unimore.it

Marco Mamei is with the Dipartimento di Scienze e Metodi dell'Ingegneria. Università di Modena e Reggio Emilia, Italy; e-mail: mamei.marco@unimore.it.

Franco Zambonelli is with the Dipartimento di Scienze e Metodi dell'Ingegneria. Università di Modena e Reggio Emilia, Italy; e-mail: franco.zambonelli@unimore.it.

In our opinion, such extreme specialization is transitory and more general-purpose approaches are likely to emerge soon. We think that future pervasive computing systems will be general purpose and users will be able to install and execute applications both on their private pervasive computing infrastructure (e.g., in smart home scenarios), and in publicly available ones (e.g., citywide infrastructures offering tourist information and services) [JonG05,Sri06]. In our opinion, this vision is motivated by the following considerations:

1. Advances in the manufacturing of pervasive computing devices (e.g., wireless sensors) will dramatically increase their performance, both in terms of computational capabilities and energy resources [Chu06].
2. Advances in energy-optimized and resource-optimized algorithms will provide efficient mechanisms to perform a number of basic services (e.g., routing), thus lowering the "resource-constraint-pressure" even further [Jon01].
3. Specialization hinders application development from a software engineering point of view. To create complex, dynamic and flexible services, it is mandatory to rely on general-purpose software infrastructures facilitating the programming task [Zam04].

All the above considerations show that general-purpose pervasive systems will be feasible in the next future, and will be required to offer advanced, flexible, robust and customizable services.

Given the extreme heterogeneity of future pervasive computing systems, their inherent dynamism and – most importantly – the incredible amount of data they will be able to produce, applications will have to autonomously adapt their behavior to different circumstances ranging from the scale of the pervasive network, to the quantity and granularity of information that will be available.

To achieve such a flexibility, applications will have to be highly context-aware (to understand and meaningfully interact with their environment) and, to this end, they will need to access properly represented contextual information.

In this direction, a number of recent researches try to represent contextual information by relying on overlay knowledge networks [Jel05, MamZ05, NagM04, Zam04]. Overlay knowledge networks can be regarded as distributed data structures encoding specific aspects of the application components' operational environment. Overlay knowledge networks are easily accessible by the components and provide easy-to-use context information (i.e., the overlays are specifically conceived to support their access and fruition).

The strength of these overlay knowledge networks is that they can be accessed piecewise as the application components visit different places of the distributed environment. This lets the components to access the right information at the right location.

From our perspective, "classic" overlay networks such as spanning tree and mesh data structures (i.e., routing distributed data structures providing components with a suitable application-specific view of the network) are particular examples of the more general concept of overlay knowledge networks [Jel05, IntG00, MadF02].

Overlay data structures such as fields and gradients [MamZ05], used in a number of macro-programming mechanisms [HadM06, NagM04], are another example of overlay knowledge network.

This paper is devoted to the above concepts and its main contribution is twofold:

1. We will better illustrate the scenario of general purpose pervasive computing showing its evolution and highlighting requirements and issues. In particular, we will discuss how considering the system as composed of a "continuum" of sensors and devices, rather than a discrete collection of them, may provide useful ideas and abstractions to deal with general purpose pervasive computing scenarios.
2. We will survey current research initiatives applying overlay knowledge networks to several autonomic and self-organizing pervasive computing applications. In particular, we will discuss how overlay knowledge networks could be suitable to the general scenario depicted above. By means of this survey, we will present how different research fields, ranging from data mining to distributed systems, are beginning to merge and complement each other to provide viable solutions to these novel scenarios.

The rest of this paper will be organized as follows. Section 2 details the upcoming scenario of pervasive computing and sensor networks, and illustrates the current shift from special-purpose and single-owner systems, to general-purpose and public pervasive infrastructures. Section 3 discusses issues and current approaches to program and gather information from pervasive distributed systems. In particular, it emphasizes the important role of overlay knowledge network in the majority of the proposals. Finally, Section 4 concludes the paper presenting some future research avenues in this area.

II. SCENARIO

As pointed out in the introduction, pervasive computing scenarios are moving toward general-purpose and widely available infrastructures that will enable a wide range of novel applications. In this section we are going to present the current setting of the scenario and its possible future evolution.

A. Current Setting

Recent advances in manufacturing and wireless communication are leading to the vision of pervasive and ubiquitous computing [But06, JonG05, Sri06]. The following

technologies, currently widespread in research labs and likely to impact soon the real world, are the workhorses of this vision:

1. **Sensor networks** consist of several micro sensors scattered across an environment that collect environmental data (e.g. sound and temperature), process data (e.g., compute average and aggregate values) and wirelessly transmit such data to other sensors or base stations. The wireless sensor networks of the near future are envisioned to consist of hundreds to thousands of inexpensive wireless nodes, each with some computational power and sensing capability, operating in an unattended mode. They are intended for a broad range of environmental sensing applications from vehicle tracking to habitat monitoring. The hardware technologies for these networks (low cost processors, miniature sensing and radio modules) are available today, with further improvements in cost and capabilities expected within the next decade [WerL06].
2. **Radio Frequency Identification (RFID) tags** are small wireless radio transceivers that can be attached unobtrusively to objects as small as a watch or a toothbrush. Tags are extremely cheap and battery-free. Thus, they do not have power-exhaustion problems. Each tag is marked with a unique identifier and provided with a tiny memory allowing to store data. Suitable devices, called RFID readers access RFID tags by radio for read or write operations. The tags respond or store data accordingly using power scavenged from the signal coming from the RFID reader [Wan06, MamZ05]. For example, a mobile device detecting tagged objects nearby can build a sort of database of the objects available. This could have several applications in inventory and ware house management [LegT06].

In our opinion, these relatively static and hard-coded applications will be soon complemented by much more dynamic ones that will leverage sensors and RFID tags as a general, publicly-available infrastructure to "interface" with the physical world. Sensor data and RFID tags will be accessed by handheld devices we carry on everyday and will provide us with information such as crowded pubs nearby, dynamically-computed bus time tables and customized and useful information about objects and products around [MamQZ06, CurG05, Bor05, NatR06]. For example, RFID tags will possibly host scripts that will enable to tell how the data in it should be handled. This can enable forms of parasitic computing (the script is executed when a reader in range powers up the tag) [Rie06]. In addition, RFID tags can be coupled with sensors. A reader can power up the sensor that takes a measure and returns it to the reader [Wan04].

3. **Localization technologies** are key enablers for pervasive computing applications. Several mechanisms and technologies are currently proposed both for outdoor and indoor localization [HigB01, Sat05]. Location in the physical world remains the primary contextual information for almost all pervasive computing applications.
4. **The Web.** Given the ever improving coverage and

bandwidth of wireless network technologies, all kind of application scenarios could benefit from the ever increasing information available on the Web. For example, it is possible can find information about the small shop round the corner and discover the menu and the price list of that nice restaurant you have seen in that little village a few days ago. Still, the Web is missing connection with the physical world and with your actual physical location. So that a query as simple such as "where is the closest Chinese restaurant?" is something that current Web cannot answer satisfactorily. There is a lot of work in this kind of location-based services, but still some general purpose architecture to implement the idea is missing [Esp01, Eag05, HarK05].

On the basis of the above considerations, future pervasive infrastructures will be hosting several services and will integrate data from various sources, ranging from RFID, sensor networks and Web resources (see Fig. 1). Users in this scenario, will be able to access – via a number of handheld and wearable devices – several services dispersed in the environment.

- Users could query, either directly or via a proper base station, sensors in the environment to get various information such as traffic reports, weather conditions, and environmental parameters (e.g., temperature, light-condition) [Bal06, Sri06].
- Users could join profile matching services and applications. Profile matching applications consist of a sensor network composed of the smart-phones of the persons joining the application (note that a Bluetooth phone can be easily regarded as a wireless sensor, in that it can provide various data to other devices around). Such sensors will monitor their surrounding environment looking for nearby "compatible" persons and notify their users upon positive matches. [Eag05].
- Users could benefit of a number of automated pervasive services to complete economic transaction and acquire information. For example, RFID allows the vision of cashier-free retailers where a user just enters a retailer, takes what he needs and, when exiting, RFID readers installed at the retailer door read the items being taken and charge the customer credit card accordingly. RFID could also allow to store information where they will be most useful. For example, information on goods and products could be stored in RFID tags stuck at that product [Bor05, NatR06].
- Users could complement and integrate all the above data and information by means of suitable Web resources. For example, a sensor network detecting some kind of polluting agent could integrate collected data with a map showing nearby industrial implants to discover possible causes of the pollution, or in a map showing natural reserves to predict dangerous effects [JRDMS]. Similarly, a group of friends could decide to share with each other their actual GPS locations, and to display them on a map which highlights pubs and bars (coming from Web-based
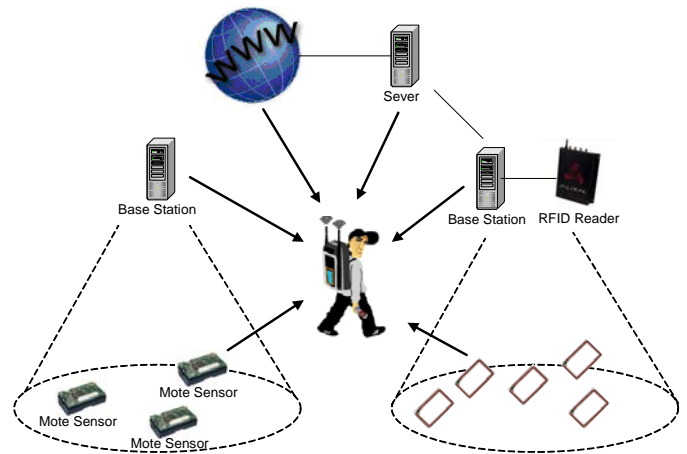
yellow pages) [Cas06].

Figure 1. General pervasive architecture

B. Future Vision

The technologies described above could lead, in the next future, to a scenario in which sensors, actuators, memory and computational infrastructures will seamlessly wrap the real world. This will allow to collect and handle data coming from an unpredictable number of devices (sensors and Web resources) that will produce a sort of enriched perception of the world. With such an infrastructure in place, several interesting applications, in which users will be able to perceive the word beyond their five senses, will become feasible. For example, while walking on a street, it will be possible to perceive (i.e., get real-time information) on how much the restaurants nearby a crowded. In a similar way, it will be always possible to "sense" where friends and relatives are located, so as to arrange for meeting on the fly.

From our perspective, there are two main streams of research fueling this vision:

1. Novel approaches are needed to provide human users and application components with "extra-sensory" information without overloading their cognitive capabilities. With regard to human users, research on wearable computer is developing mechanisms to enable a person to see (by means of suitable see-through visors) computer-generated images overlaid to the physical world. Such images can augment the word by providing additional information [Dan06]. For example, they could show directions overlaid to the actual environment, or provide personal information overlaid to the person we are actually talking with. With regard to application components, suitable software infrastructures are needed to represent context information in a way that will be easy for the components to understand and use [MamZ05].
2. It is fundamental to actually store and manage that information at the infrastructure level. Research on RFID tags and sensors infrastructures, is a promising approach (complementary to the previous one) leading to this vision. In this context, the idea is to store and later retrieve information in the RFID tags and sensors that are likely to

populate (and saturate) our physical environment. Such an infrastructure could be used to enrich the world with context information that could be retrieved properly [MamQZ06]. For example, the infrastructure would allow to store "virtual" post-it notes across an environment to be found later on.

It is rather clear that such a vision implies a huge amount of information and data pervading the physical world that (given its scale) requires novel methodologies to be dealt with. In our opinion, a paradigm leading to the development of proper methodologies, in this context, could be based on the "continuum" abstraction [BeaB06]. Following this approach the system is designed having in mind a continuum of data sources (rather than a discrete network of devices) and so the abstraction being realized have to scale to an arbitrary number of devices. Of course to deal with such kind of large scale systems, autonomic and self-organization principles are needed [Dob07]. This is because managing the system at a fine-grained scale and addressing individual components will not be feasible (with the continuum abstraction in mind, the very concept of individual component tend to vanish), and so autonomic and self-organization mechanisms – where individual components manage themselves -- have to be introduced.

In particular, we envision an architecture, like the one depicted in Fig. 2. There, a countless number of sensors (wireless mote sensors, RFID, smart phones, and yet-to-come devices) enrich the world with digital information. This layer (represented as the bottom layer in Fig. 2) will be constituted by a huge number of heterogeneous and dynamically varying devices. The data at this basic level is at the finest possible granularity, and because of that will be hardly manageable and understandable by application components (i.e., too much data, too sparse knowledge).

Overlay knowledge networks are distributed data structures encoding specific aspects of the application components' operational environment. Overlay knowledge networks are easily accessible by the components and provide easy-to-use context information [MamZ05]. These overlay knowledge networks come into play to organize the data of the bottom layer into higher-level and more semantically expressive concepts. An example of this idea would be an overlay knowledge network that aggregates the data produced in a region of the underlying network to offer application components a single aggregated value (e.g., the average) representing the whole region. In other words, data produced by the bottom sensors can be aggregated at different level of abstractions. This aggregation produce discrete data elements each one managing portions of the continuum sensor space. These elements of the overlay knowledge network are represented in the higher layers of Fig. 2 and the upward arrows represent the process of creating higher-level concepts from low-level sensors.

This upward direction is not the only possible. In several situation, overlay knowledge network need to integrate and contextualize high-level concepts to a lower layer using sensor data. This integration is represented by the bottomward arrows in Fig. 2.

The resulting scenario is that of a hierarchy of an arbitrary number of overlays representing context information at different level of granularity. Application components, depending on their task, decide at which level to consider the context. Lower-level information will be aggregated to the proper level of abstraction. Higher-level information will be possibly contextualized to that level, and all this information will be integrated together in coherent view supporting application tasks.

Although the above description is at the level of modeling, and data aggregation, contextualization and integration mechanisms could be realized via whatever approach, in practice the model easily support a hierarchical architecture where higher-level servers collect and provide data at a certain level of granularity. Adopting this viewpoint, at the top level of Fig. 2, we have globally accessible Internet server providing worldwide aggregated information. At the lower layers, there are servers providing more and more specific data (e.g., state-wide, city-wide, building-wide data). At the bottom-layer there are the individual sensors offering extremely localized – but extremely detailed and up-to-date -- information.

Whatever the architecture, in order to realize the conceptual model in Fig. 2, it will be fundamental to rely on self-organization and autonomic principles. In fact, to guarantee robustness and scalability, the overlay knowledge network will have to maintain its coherency despite network glitches, sensors failures, the addition and removal of part of knowledge and other kind of contingencies.
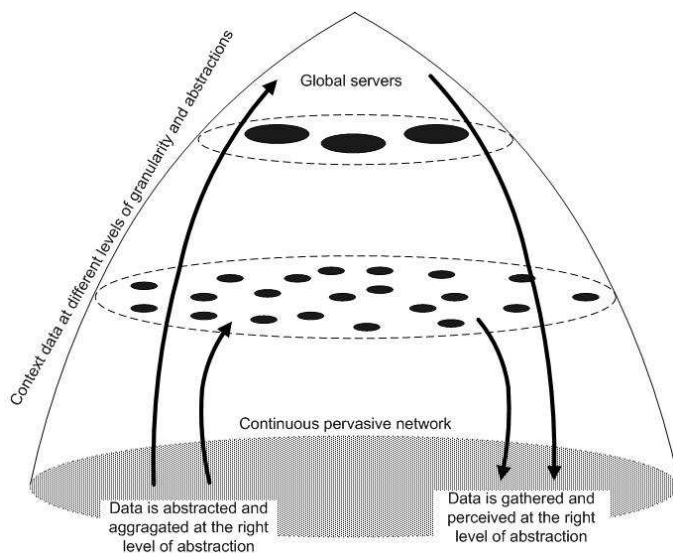

Figure 2. Continuum pervasive network with an arbitrary number of overlays describing context at different granularity

III. ISSUES AND CURRENT APPROACHES

Several new technologies and mechanisms are needed to fulfill the above vision and to create general purpose pervasive applications. In particular, we think that the main challenge is to provide applications with suitable overlay

knowledge networks to gather, understand and exploit context information at the proper level of abstraction for their application task. If a suitable context-representation is available, often the application task becomes easy, since application components see clearly from their context how to achieve the task [MamZ05].

From our perspective, there are three main research fields that are fruitfully tackling the above problems by exploiting overlay knowledge networks.

1. **Data Retrieval and Aggregation** comprises a number of researches trying to get data from a distributed sensors in an efficient way. In this context, overlay knowledge networks are used to create the routing structures to collect and aggregate data.

2. **Macro Programming** deals with programming a distributed system without explicitly defining single entities activities, but letting a compiler or a distributed middleware to translate high-level task into individual component activities in an automatic way. In this context, overlay knowledge networks are used to create regions and areas in a distributed systems allowing to suitably differentiate application execution disregarding individual components' activities.

3. **Data Integration** allows to integrate data from various sources (Web services and pervasive sensors) to offer application components an all-encompassing view of the operational environment (context). In this context, overlay knowledge networks are used to actually represent the integrated view that will be provided to application components.

In the next subsections we will present a survey of current research initiatives in these areas, showing also how the different areas themselves complements one another and pursue from different perspective the same ultimate goals.

A.   Data Aggregation and Retrieval

The main goal of a sensor network (and of the majority of pervasive computing systems) is to collect data from the environment and to suitably present the data to application components. For this reason several researches try to devise mechanism to retrieve, collect and possibly aggregate data form a sensor network. The most common approach to collect data from the network consists in deploying data collector (i.e., sink) nodes which subscribe to some type of data flowing from sensing nodes about some particular phenomena. Once a data collector is registered to the network, each node starts to periodically send data to it. For example there may be a sink interested in receiving data from a particular region "A" between 2pm and 6pm if the temperature in that zone exceed 50°. Each day, during the selected time frame, sensors which detect temperatures over the selected threshold will send data to the sink. This is the simplest possible approach to retrieve data but has several disadvantages. In general since different sensor nodes detect the same phenomenon, it is likely that there will be an high degree of redundancy in the data flowing to the sink from different sources. Moreover each node located between a source and sink has to spend energy to

route the message towards the destination. When compared to local processing of data, wireless transmission is extremely expensive. Researchers at the University of California, estimate that sending a single bit over radio is at least three orders of magnitude more expensive than executing a single instruction [ShrP04]. Last but not least, this approach is very sensitive to reading errors and sensors faults. If a node, broken or malicious, produces fake data, there is no straightforward way to filter it out.

To overcome the above problems, in-network filtering, processing and aggregation techniques can be used to conserve the scarce energy resources and improve data quality. From the information sink point of view in network data aggregation has two main advantages. The first one consist on a reduction of the potentially overwhelming data streams produced by the sensors. The second one, due to the activity of filtering and processing, is to reduce the complexity and the amount of data gathered letting further analysis more manageable. Probably, during the next few years, due to the increase of the size and density of sensor networks these advantages will quickly become determinant and every application will use some mechanisms where some sort of "in network" aggregation will be implemented natively.
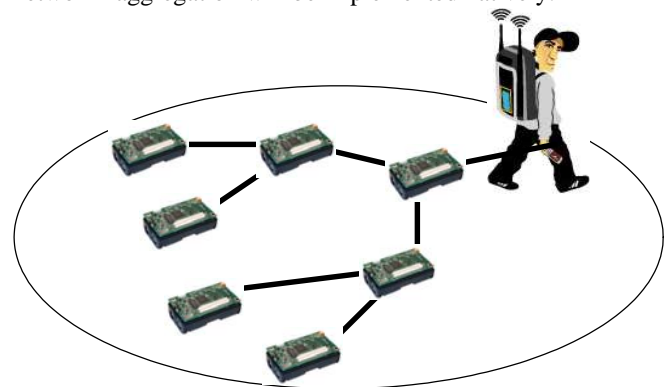

Figure 3. A spanning tree is created in the sensor network to route the collected data to a root node.

The work described in [Jel05] distinguishes *reactive* and *proactive* protocols for computing aggregate functions in a sensor network.

- *Reactive protocols* try to respond on demand to queries injected by nodes. If the answer is found in some region of the network, it is routed directly to the issuer node (see Fig. 3). Examples of this approach are well described in [IntG00, MadF02].

- *Proactive protocols* continuously provide aggregated data using some function and aim to diffuse meaningful values on every nodes in the networks in an adaptive way (see Fig. 4). "Adaptive" means that if sensed values change over time, the output of the algorithm should track variations reasonably quickly. Proactive protocols are often useful when aggregation is used as a building block for completely decentralized solutions to complex tasks [Jel05].

The above computation of aggregate functions is a key building block for many applications. In fact, aggregate data can be regarded as a simplified view of the components operational environment. Components may find simpler to access the aggregate value rather than distill the individual sensor readings.

Some examples of most used aggregated values are network size, average load, average uptime, location and description of hot spots, and so on. Local access to global information is often very useful, if not indispensable for building applications that are robust and adaptive. For example a fire alarm system has to trigger an alarm if the average temperature inside a building exceed a certain threshold or a distributed storage system has to know the overall free space over various device before processing a write() request. To reach the goal of a local access to global network features we have mainly two choices.

- The first one consists of gathering on some sinks all the (aggregated on not) sensor readings. After that we have to diffuse the global aggregated values into the overall network. This approach is simple and straightforward but has several serious limitations. The main one is the poor scalability. In fact as the network size grows, the amount of data that the sink has to manage become quickly overwhelming.
- On the other side we can use gossip based aggregations methods [Jel05]. Using this kind of algorithm local sensor readings are not to be convoyed to a sink, but can stay on sensors. The core of these protocols is a simple gossip-based communication scheme in which each node periodically selects some other random node to communicate with. During this communication the nodes update their local approximate values by performing some aggregation specific and strictly local computation based on their previous approximate values. After some iterations the local approximate value converge to the global value. The main advantages of these methods are that they are simple, scalable and provide local access to global values without any additional burden.

The last reported feature is really important in our vision. In a world full of sensors and actuators, users will need simple (i.e., aggregated) representations of the area of the network where they will be immersed. Using traditional routing based aggregation algorithm, due to their inherent "reactive" nature, will require, for each query, the building of a dedicated tree and to wait answers from an unknown number of sensors (which will may be very high). Instead, using gossip based algorithm, any user will be able to get, without any additional burden for the network, a simplified view of the area.

In general, the resulting aggregate value distributed across the network becomes an instance of overlay knowledge network. The overlay in fact extracts low level sensor reading to higher level concepts (i.e. aggregate values).
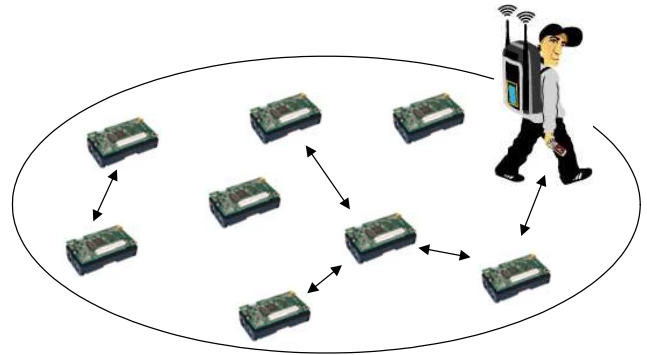

Figure 4. A gossip algorithms is run by nodes to aggregate data and report them back to an inquiring node.

In the next paragraph we briefly highlight some general examples of either reactive and proactive algorithm applications.

Data aggregation and retrieval is at the basis of a number of relevant application in the context of pervasive computing and sensor network. Currently the main application of sensor networks is environmental monitoring. This application consist of deploying a suitable number of ad hoc wireless connected sensors in a region. Such devices periodically read some environmental properties and route the acquired data towards a base station that is in charge of gathering and storing them. A good example of this kind application has been deployed on a natural reserve island in front of the Maine coast [Pol06]. There a hundred of sensors collect data from the birds nest, monitoring their micro climate. The data being collected are sent over the Internet and publicly available over the web.

Another promising application, which has not yet been fully developed, is object tracking. This activity consists of recognize and subsequently track moving targets over a monitored field. To achieve this task sensors do not have to collect massive amount of data to a central station for further analysis, but the network have to process sensed information and produce a simplified view of the physical world in which the object being tracked is readily visible. This application has been originally conceived in the military setting to drive vehicles in un-trusted areas. A promising new approach of this application involves multi sensory tracking. With this mechanism the same phenomenon can be recognized by means of different sensory inputs. For example, a car reaching a blind spot in a camera network could be tracked using sound sensors.

B. Macro Programming

A key challenge in pervasive computing is to provide powerful programming models to facilitate the development of applications in dynamic and heterogeneous environments.

One of the main conceptual difficulties is that we have direct control only on the agents' local activities, while the application task is often expressed at the global scale [Zam04]. Bridging the gap between local and global activities is not easy, but it is possible: distributed algorithms for autonomous sensor networks like the ones presented in the previous subsection have been proposed and successfully

verified, routing protocols is MANET (in which devices coordinate to let packets flow from sources to destinations) have already been widely used. The problem is still that the above successful approaches are ad-hoc to a specific application domain and it is very difficult to generalize them to other scenarios.

One promising research initiative in this direction is macro programming. The idea is to specify the global application tasks to be achieved and leaving to a compiler or a distributed middleware [HadM06, Nag02, NagM04] the tasks of mapping these global task into individual component activities. To build these languages there are two fundamental challenges:

- devise a global language suitable for a relevant class of applications
- devise a set of distributed algorithms to map the language into the component activities.

The above two tasks aim at hiding from the programmer low level details such as the heterogeneity and the scale of the underlying network.

In the last few years a number of research initiatives addressing macro programming have been proposed in several application scenarios.

In the Amorphous Computing project [Nag02], a macro-programming language is used to control shape formation in a reconfigurable sheet composed of thousands of identically-programmed, locally-interacting robotic agents. The desired global shape is specified at an "abstract" level as a folding construction on a continuous sheet of paper (i.e., origami). This construction is then automatically compiled to produce the program run by the identically-programmed agents. The global language allows to define the regions where the sheet has to fold, leaving to the compiler the identification of the low level action needed to actually reconfigure (i.e., bend) the robots.

Similar approaches for the control of shape and motion in a modular robot (i.e. a collection of simple autonomous actuator with few degrees of freedom connected with each other) have been recently proposed [StoN04, WerB06]. In these approaches a global description of the shape to be formed or of the gait to be followed is provided to the robot, either by representing the shape in some coordinate frame, or by adopting a description functionally specifying how the robot has to bend its actuators to move. Such a global description is then compiled into low level messages and actions to drive and coordinate the individual modules.

TinyDB [MadF02] and Cougar [YaoG02] provide a high-level SQL or XML-based query interface to sensor network data. The query is expressed by means of a high-level language indicating the data to be gathered in a declarative way. A compiler translates the query into the low-level sensor activities needed for the creation of the proper data collection and aggregation distributed algorithms.

Spatial Programming (SP) [Bor04] is a macro programming approach to program a sensor network. This approach allows to define regions in the network adopting a high-level semantic. In SP, for example, it is possible to address (and get a handle to) all the sensor in a given geographic region (described e.g. by its latitude and longitude). A low-level distributed middleware in then in charge to set-up suitable routing structures to actually address the proper sensors.

Abstract Regions (AR) [NewA04] is another macro programming approach to define regions in a sensor network. Rather than focusing on geographic regions like in Spatial Programming, AR focus on network regions (e.g., x-hop neighbors, spanning tree and planar meshes). A high-level language allows to specify the network region, while low level algorithm create the actual routing structure to handle the proper nodes.

Regiment is a functional macro programming [WelN04] language that generalize both the previous approaches. Regiment allows to define regions in the network able to represent spatially distributed, time-varying collections of node state. The programmer uses the language to express interest in a group of nodes with some geographic, logical, or topological relationship, such as all nodes within k radio hops of some anchor node. A distributed middleware is then in charge to map the regions into suitable sensor-level coordination protocols. Similar approaches to define regions in a distributed system according to spatial and functional characteristics have been presented in [BecH04]

A more comprehensive survey of currently proposed macro-programming languages can be found in [HadM06].

In general, all the reported macro-programming approaches uses suitable overlay knowledge networks to control the distributed program. In most of the proposals, overlay knowledge networks are used to define the regions where the components activities will be different. In Spatial Programming, for example, the overlay knowledge network is represented by the data structure identifying the region where data should be collected by the application.

To create complex, dynamic and flexible services, it is mandatory to rely on general-purpose software infrastructure facilitating the programming task. The ability to program a distributed system without explicitly and directly defining individual entities' activities will be a fundamental asset in this direction.

C. Data Integration

Pervasive computing applications will be naturally integrated with Web services and Internet resources. Not only Web services will be a natural technology to access pervasive applications remotely, but it could also provide further context information to the pervasive device. For example, sensors could get from the Internet the average temperature of the region they are in, and compare their sensor readings with that average. With this regard, we think that in the next future application will integrate together data coming from the Internet and data coming from the real world (sensors) and actually merge it together in a coherent framework providing advanced context-aware applications.

In this context, overlay knowledge networks are used to merge the collected data together, and to provide such data to application components in a coherent view.

A number of recent projects from different research communities (data mining, distributed systems, semantic Web,

47

Web services, etc.) are tackling the challenge of data integration across multiple providers.

One interesting research in this area is described in [PerP04]. The goal of this project is to develop a context-awareness system to detect and infer domestic activities performed by the users. The proposed approach is to infer the activities of the user on the basis of the objects he touches. For example, by sensing that the user touches a "teapot", some "teabags", "glasses" and "spoons", the system can infer that the user's action is "making tea". This kind of knowledge could be of use in a number of smart-home scenarios. To implement such an idea, the system relies on RFID tags associated to (and identifying) everyday objects, and gloves integrated with RFID reader worn by the user. This allows the system to detect, rather naturally, what the user is touching.

This stream of data coming from pervasive devices requires models of activities to detect what the user is doing. Such models are automatically mined from the Web. In particular, the system connects to specific "How to" sites, describing how to perform a specific activity, extracts the labels associated to the object being used, and creates a Bayesian network describing probabilistically the objects involvement in the different activities. The model is finally, checked against the data coming form the RFID reader to infer the activities being carried on.

In our opinion, this project is a perfect example of the fact that pervasive and Web resources complement each other, and by integrating them, it is possible to obtain novel and powerful services.

Another relevant approach is presented in [Eag03]. The goal of this work is to infer users context by capturing their speech. The voice of the user is record by a PDA carried on by the user. The voice signal is sent over a wireless network to a server that process the signal and transcribes the speech. The server connects to a Web service called Concept Net [Liu04] that is based on a knowledge network describing common-sense activities. Concept Net is, in fact, a huge repository of commonsense sentences (e.g., you'd order food in a restaurant) and a suitable API to access and mine the repository.

By providing ConceptNet with the speech transcription, the service is able to infer the most likely context for the user. For example, the speech: "Hi, today I'm going to have a cheeseburger and a beer" would let ConceptNet infer that the user context is "ordering food at a resturant". Such information is then sent back to the PDA for further actions.

Another interesting mechanism to combine sensor data and Web information involves the usage of GPS as sensors and Web-retrieved maps from open GIS-tool like Google Earth (http://earth.google.com). In [Cas06], we describe two services in this direction. A first service allows a user equipped with a RFID reader and a GPS device to see his actual location and past movements, and to dynamically create Google Earth placemarks of the tagged objects being read with the RFID reader at the right location. This service can be fruitfully employed in a number of situations. In particular, we focused on the scenario in which a tourist wants to automatically build and maintain a diary of his journey. To this end, the proposed service allows to keep track of all the user movements and have them displayed on the map of the visited place. Moreover, the support for RFID allows to access likely-to-be-soon-available tourist information stored in RFID tags attached to art-pieces. From the diary perspective, this allows to store the visited art-pieces' location together with their description on the journey map. In addition, our service could also provide with important logistic information. For example, the action of reading the tag of the user's car at a certain location triggers a new car-placemark on Google Earth showing the actual position of the car. This allows the tourist to easily recall where the car has been parked.

Another service, allows multiple users to share their list of placemarks and their current location. Again, this service can be employed in several scenarios, and we focused on supporting a group of tourists cooperatively visiting a place. Such a situation applies to a class of students or to a group of boy-scouts, where each person can visit the place independently, while keeping in touch and sharing information with the other members. To this end the service allows to share GPS data with other members and with the group leader (e.g., the teacher may be in need of monitoring the location of all the students). Moreover, placemarks pointed by one person may be shared across al the group. This can be useful to share opinions or interesting sightings, but also to easily agree on some meeting points. For example, by sharing placemarks, all the users can spot a suitable place (e.g., a pub) that is in the middle of them and agree to meet there (see Figure 5).

Other approaches in this direction, developed by other research groups, [PatL04] combine GPS data and maps to create a probabilistic model of the user activities. This approach allows to the system to learn the user motion routine (e.g., where does he go, where does he park the car, etc.) and possibly to check anomalies against the learned trend.
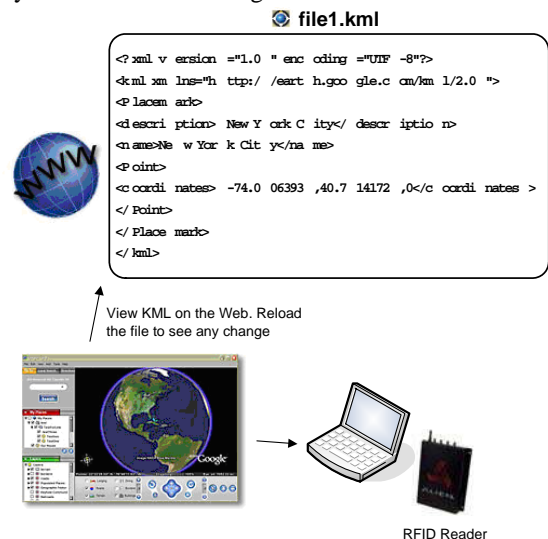

Figure 5. Integration of GPS data and Web maps.

Finally, another source of information that researchers are trying to integrate is that coming from images widely available and tagged by services like Flickr (www.flickr.com). The idea at the core of some recent researches is to try to match pictures taken from cameras with those available on the

Internet. This would allow to get information about objects without the need of tagging them artificially. For example, the image of a tower taken by a camera phone could be matched against a data base of images to properly recognize it as the Pisa leaning tower [Jia06].

All the above examples show rather clearly that the approach of integrating resources and data from pervasive systems and Web resources in a promising research avenue.

IV. CONCLUSIONS

In this paper we presented our vision for next future pervasive computing systems. In our opinion, these systems will be general purpose and users will be able to install and execute applications both on their private pervasive computing infrastructure (e.g., in smart home scenarios), and in publicly available ones (e.g., citywide infrastructures offering tourist information and services). Given the extreme heterogeneity of this scenario, its inherent dynamism and – most importantly – the incredible amount of data the system will be able to produce, applications will be required to match and comply those characteristics. Applications will have to autonomously adapt their behavior to different circumstances ranging from the scale of the pervasive network, to the privacy-level being requested by the users. To achieve such a flexibility applications will have to be highly context-aware (to meaningfully interact with their environment) and autonomic. To this end, they will be able to gather relevant context information both from the pervasive network sensing the environment and from global-accessible Internet services. We also introduced how considering the system as composed of a "continuum" of sensors and devices, rather than a discrete collection of them, may provide useful ideas and abstractions to deal with the above challenges.

In addition, we presented the key mechanisms and researches trying to fulfill the above vision:

- Retrieve and aggregate data will provide developers with advanced tools to get data from a distributed system in an efficient way.
- Macro Programming a distributed system deals with programming a distributed system without explicitly defining single entities activities, but letting a compiler or distributed middleware to translate high-level task into individual component activities. This will allow developers to design systems composed of a huge number of components that will be able to carry on complex coordinated activities.
- Integrate data gathered from various sources allows to offer application components a coherent view of their context.

In particular, we tried to present how the concept of overlay knowledge networks may be at the basis of most of the proposal, and how overlay knowledge network may represent a framework to develop applications in future pervasive computing scenarios.

In our opinion, these researches are only at the beginning of addressing satisfactorily the requirements of future scenarios and several questions remain open: How to represent context information in a general way? How can we retrieve and access such huge amount of knowledge? Which kind of autonomic algorithms should we enforce to add robustness and self-organization properties to those systems?

Our future research within the CASCADAS European project will try to address some of these questions.

REFERENCES

[Bal06] H. Balakrishnan, S. Madden, V. Bychkovsky, K. Chen, W. Daher, M. Goraczko, H. Hu, B. Hull, A. Miu, E. Shih, "CarTel: A Mobile Sensor Computing System", MIT CSAIL Report, 2006.

[BeaB06] J. Beal, J. Bachrach, "Infrastructure for Engineered Emergence on Sensor/Actuator Networks", IEEE Intelligent Systems, 21(2):10-19,2006

[BecH04] C. Becker, M. Handte, G. Schiele, K. Rothermel, "PCOM - A Component System for Pervasive Computing", IEEE International Conference on Pervasive Computing and Communications, Orlando (FL) ,USA , 2004.

[Bor04] C. Borcea, D. Iyer, P. Kang, A. Saxena, L. Iftode, "Spatial Programming using Smart Messages: Design and Implementation", International Conference on Distributed Computing Systems, Tokio, Japan, 2004.

[Bor05] G. Borriello, "RFID: Tagging the World", Communication of the ACM, ACM Press, 48(9):34-79, 2005.

[But06] D. Butler, "2020 Computing: everything, everywhere", news@nature.com, 22 March 2006.

[Cas06] G. Castelli, M. Mamei, A. Rosi, F. Zambonelli, "Browsing the World: bridging pervasive computing and the Web", International Workshop on Ubiquitous Geographical Information Services, Munster, Germany, 2006.

[Chu06] C. Park, P. Chou, Y. Sun, "A Wearable Wireless Sensor Platform for Interactive Art Performance", IEEE International Conference on Pervasive Computing and Communications, Pisa, Italy, 2006.

[CurG05] C. Curino, M. Giani, M. Giorgetta, A. Giusti, A. Murphy, G. Picco, "TinyLIME: Bridging Mobile and Sensor Networks through Middleware". IEEE International Conference on Pervasive Computing and Communications, Kauai Island, HW (USA), 2005.

[Dan06] W. Daniel, T. Pintaric, F. Ledermann, S. Dieter, "Towards Massively Multi-User Augmented Reality on Handheld Devices", International Conference on Pervasive Computing, Munich, Germany, 2005.

[Dob07] S. Dobson, S. Denasiz, A. Fernandez, D. Gaiti, E. Gelenbe, F. Massacci, P. Nixon, F. Saffre, N. Schmidt, F. Zambonelli, "A Survey of Autonomic Communications", ACM Transactions on Autonomous and Adaptive Systems, ACM Press, to appear, 2007.

[Eag03] N. Eagle, P. Singh, and A. Pentland, "Common sense conversations: understanding casual conversation using a common sense database". Workshop on Artificial Intelligence, Information Access, and Mobile Computing, Acapulco, Mexico, 2003

[Eag05] N. Eagle and A. Pentland, "Social Serendipity: Mobilizing Social Software", IEEE Pervasive Computing, IEEE CS Press, 4(2):28-34, 2005.

[Esp01] F. Espinoza, P. Persson, A. Sandin, H. Nystrom, E. Cacciatore, M. Bylund, "GeoNotes: Social and Navigational Aspects of Location-Based Information Systems", International Conference on Ubiquitous Computing, Atlanta (GE), USA, 2001.

[HadM06] S. Hadim, N. Mohamed,"Middleware Challenges and Approaches for Wireless Sensor Networks", IEEE Distributed Systems Online, 7(3), 2006.

[HarK05] R. Hariharan, J. Krumm, E. Horvitz, "Web-Enhanced GPS", International Workshop on Location and Context Awareness, Munich, Germany, 2005.

[HigB01] J. Hightower, G. Borriello, "Location Systems for Ubiquitous Computing", IEEE Computer, IEEE CS Press, 34(8):57-66, 2001.

[IntG00]C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor

networks", International Conference on Mobile Computing and Networking, Rome, Italy, 2000.

[Jel05] M. Jelasity, A. Montresor, O. Babaoglu, "Gossip-Based Aggregation in Large Dynamic Networks", ACM Transaction on Computer Systems, ACM Press, 23(3):219-252, 2005.

[Jia06] M. Jia, X. Fan, X. Xie, M. Li, W. Ma, "Photo-to-Search: Using Camera Phones to Inquire of the Surrounding World", International Conference on Mobile Data Management, Nara, Japan, 2006.

[Jon01] C. Jones, K. Sivalingam, P. Agrawal, J. Chen, "A Survey of Energy Efficient Network Protocols for Wireless Networks", Wireless Networks, Kluwer Academic Publisher, 7(4):343-358, 2001.

[JonG05] Q. Jones, S. Grandhi, "P3 Systems: Putting the Place Back into Social Networks", IEEE Internet Computing, IEEE CS Press, 9(5):38-46, 2005.

[JRDMS] James Reserve Data Management System, http://dms.jamesreserve.edu

[LegT06] C. Legner, F. Thiesse, "RFID-Based Facility Maintenance at Frankfurt Airport", IEEE Pervasive Computer, IEEE CS Press, 5(1):34-39, 2006.

[Liu04] H. Liu, P. Singh, "ConceptNet: A Practical Commonsense Reasoning Toolkit", BT Technology Journal, Kluwer Academic Publishers, 22(4):211-226, 2004.

[MadF02] S. Madden, M. Franklin, J. Hellerstein, W. Hong. "TAG: A Tiny AGgregation Service for Ad-Hoc Sensor Networks", Symposium on Operatign Systems Design and Implementation, Boston (MA) ,USA, 2002.

[MamQZ06] M. Mamei, R. Quaglieri, F. Zambonelli, "Making Tuple Spaces Physical with RFID Tags", ACM Symposium on Applied Computing, Dijon, France, 2006.

[MamZ05] M. Mamei, F. Zambonelli, "Physical Deployment of Digital Pheromones Through RFID Technology", IEEE Swarm Intelligence Symposium, Pasadena (CA), USA, 2005.

[Nag02] R. Nagpal, "Programmable Self-Assembly Using Biologically-Inspired Multiagent Control", Joint Conference on Autonomous Agents and Multi-Agent Systems, Bologna, Italy, 2002.

[NagM04] R. Nagpal, M. Mamei, "Engineering Amorphous Computing Systems", Methodologies and Software Engineering for Agent Systems, Kluwer Academic Publishing, 2004.

[NatR06] B. Nath, F. Reynolds, R. Want, "RFID Technology and Appliations", IEEE Pervasive Computing 5(1):22-69, 2006.

[NewA05] R. Newton, Arvind, M. Welsh, "Building up to Macroprogramming: An Intermediate Language for Sensor Networks", International Symposium on Information Processing in Sensor Networks, Los Angeles (CA), USA, 2005.

[PatL04] D. Patterson, L. Liao, K. Gajos, M. Collier, N. Livic, K. Olson, S. Wang, D. Fox, H. Kautz, "Opportunity Knocks: a System to Provide Cognitive Assistance with Transportation Services", International Conference on Ubiquitous Computing, Tokyo, Japan, 2004.

[PerP04] M. Perkowitz, M. Philipose, D. Patterson, K. Fishkin. "Mining Models of Human Activities from the Web". International World Wide Web Conference, New York (NY), USA, 2004.

[Pol06] J. Polastre, R. Szewcyk, A. Mainwaring, D. Culler, J. Anderson, "Analysis of Wireless Sensor Networks for Habitat Monitoring", Wireless Sensor Networks, Kluwer Academic Publishers, 2004.

[Rie06] M. Rieback, B Crispo, A. Tanenbaum, "Is Your Cat Infected with a Computer Virus?", IEEE International Conference on Pervasive Computing and Communications, Pisa, Italy, 2006.

[Sat05] I. Satoh, "A Location Model for Pervasive Computing Environments", Proceedings of IEEE International Conference on Pervasive Computing and Communications, Kauai Island, HW (USA), 2005.

[ShrP04] Nisheeth Shrivastava, Chiranjeeb Buragohain, Divyakant Agrawal and Subhash Suri, "Medians and Beyond: New Aggregation Techniques for Sensor Networks," ACM Conference on Embedded Networked Sensor Systems (SenSys 2004), 2004.

[Sri06] M. Srivastava, M. Hansen, J. Burke, A. Parker, S. Reddy, G. Saurabh, M. Allman, V. Paxson, D. Estrin, "Wireless Urban Sensing Systems", CENS Technical Report #65 , 2006.

[StoN04] K. Stoy, R. Nagpal, "Self-Reconfiguration Using Directed Growth", International Symposium on Distributed Autonomous Robotic Systems", Toulouse, France, 2004.

[Wan04] R. Want, "Enabling Ubiquitous Sensing with RFID", IEEE Computer, IEEE CS Press, 37(4):84-86, 2004.

[Wan06] R. Want, "An Introduction to RFID Technology", IEEE Pervasive Computer, IEEE CS Press, 5(1):25-33, 2006.

[WelN04] M. Welsh, R. Newton, "Region streams: functional macroprogramming for sensor networks", International Workshop on Data Management for Sensor Networks, Toronto, Canada, 2004.

[WerB06] J. Werfel, Y. Bar-Yam, D. Rus, R. Nagpal, "Distributed construction by mobile robots with enhanced building blocks", International Conference on Robotics and Automation, Orlando (FL), USA, 2006.

[WerL06] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, M. Welsh, "Deploying a Wireless Sensor Network on an Active Volcano.", IEEE Internet Computing, IEEE CS Press, 10(2):18-25, 2006.

[YaoG02] Y. Yao, J. Gehrke, "The Cougar approach to in-network query processing in sensor networks", ACM Sigmod Record, ACM Press, 31(3):9-19, 2002.

[Zam04] F. Zambonelli, V. Parunak, "Towards a Paradigm Change in Computer Science and Software Engineering: a Synthesis", The Knowledge Engineering Review, 18(4), 2004.