

Chapter 2

Introduction to Cooperating Objects

A number of different system concepts have become apparent in the broader context of embedded systems over the past couple of years. First, there is the classic concept of **embedded systems** as mainly a control system for some physical process (machinery, automobiles, etc.). More recently, the notion of pervasive and **ubiquitous computing** started to evolve, where objects of everyday use can be endowed with some form of computational capacity, and perhaps with some simple sensing and communication facilities. However, most recently, the idea of **Wireless Sensor Networks** has appeared, where entities that sense their environment not only operate individually, but collaborate together using ad hoc network technologies to achieve a well-defined purpose of supervision of some area, some particular process, etc.

We claim that these three types of systems (i.e. embedded systems, pervasive and ubiquitous computing and wireless sensor networks) that act and react on their environment are actually quite diverse, novel systems that, on the one hand, share some principal commonalities and, on the other hand, have some different aspects that complement each other to form a coherent group of objects that cooperate with each other to interact with their environment. In particular, important notions such as control, heterogeneity, wireless communication, dynamics/ad-hoc nature, and cost are present to various degrees in each of these types of systems.

Fig. 2.1 shows the different weights of these functional aspects. As already mentioned, the strength of traditional embedded systems is control functionality. Pervasive computing applications include control aspects as well but usually do not have hard real-time constraints. Heterogeneity is a key aspect of pervasive computing since no common platforms can be assumed if devices of everyday's life are cooperating. In contrast, single embedded systems and wireless sensor networks are a controlled setting where heterogeneity is typically low. If several embedded systems are combined as, for example, in a car, their heterogeneity increases. A characteristic of wireless sensor networks and pervasive computing is wireless communication whereas in contrast traditional embedded systems are wired. This has direct implications on dynamics and ad-hoc nature since wired systems are static. The dynamic nature of pervasive computing is tightly related with its heterogeneity. For wireless sensor networks there exist both static and mobile scenarios. Since many of them assume a high number of sensor nodes low cost is very important. If many devices of our environment should be integrated in pervasive computing applications the cost aspect will become more relevant especially for cheap devices. Since embedded systems are usually integrated into larger and more expensive devices the cost of the single embedded system is less important than for the other system concepts.

The conception of a future-proof system would have to combine the strong points of all three system concepts at least in the following functional aspects:

- Support the control of physical processes in a similar way embedded systems are able to do today.

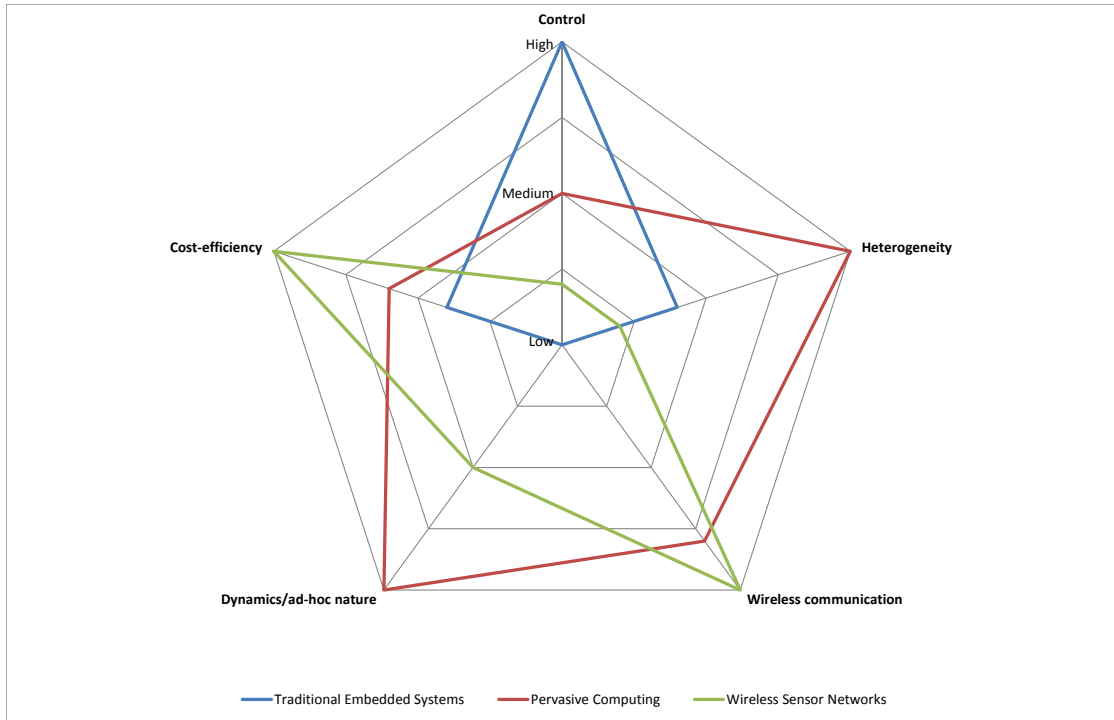


Fig. 2.1 Strengths/Importance of key functional aspects in different system concepts

- Have as good support for device heterogeneity and spontaneity of usage as pervasive and ubiquitous computing approaches have today.
- Be as cost efficient and versatile in terms of the use of wireless technology as Wireless Sensor Networks are.

The convergence of these three types of technologies that, until now, have been evolving independently of each other (Fig. 2.2), is what we call Cooperating Objects technologies. This new term is born out of the combination of these traditional systems.

Moreover, this notion or paradigm of Cooperating Objects is even stronger than the individual technologies it stems from, as it carries over to their internal structure – e.g. a Wireless Sensor Network can be regarded as consisting of Cooperating Objects itself, highlighting the diversity of cooperating patterns admissible under this general paradigm. Also, pointing to the importance of complementing the vision of pervasive computing with that of pervasive control is essential.

2.1 Definition

Following the concepts we have just discussed, let us now define more formally what a Cooperating Object is:

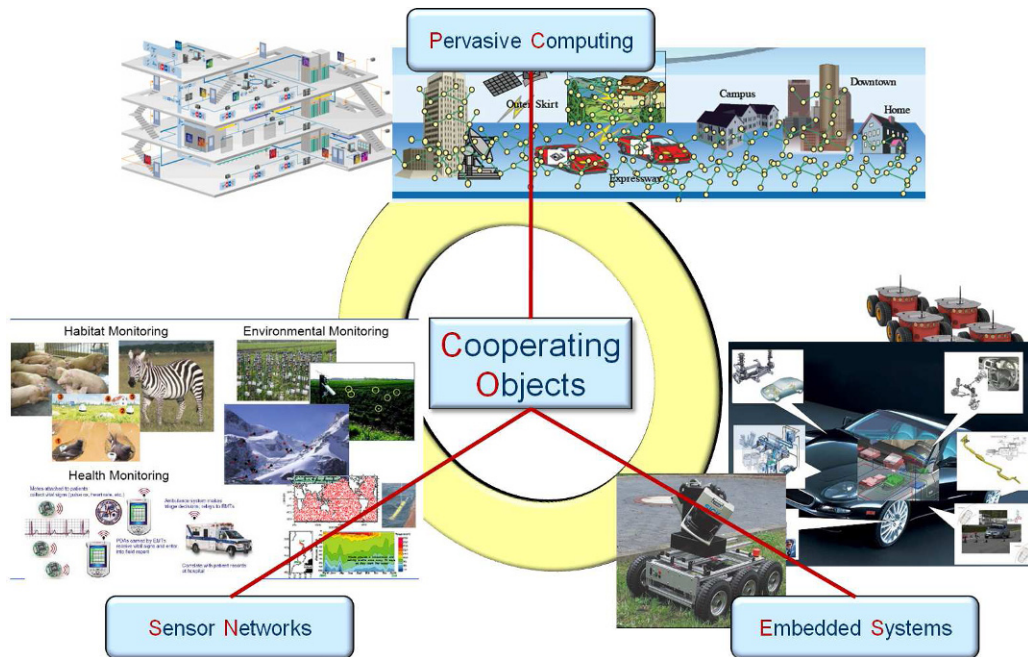


Fig. 2.2 Areas covered by Cooperating Objects

“Cooperating Objects consist of embedded computing devices equipped with communication as well as sensing or actuation capabilities that are able to cooperate and organize themselves autonomously into networks to achieve a common task. The vision of Cooperating Objects is to tackle the emerging complexity by cooperation and modularity. Towards this vision, the ability to communicate and interact with other objects and/or the environment is a major prerequisite. While in many cases cooperation is application specific, cooperation among heterogeneous devices can be supported by shared abstractions.”

According to this definition, computing devices are the core of Cooperating Objects. They interact with their environment either by monitoring it (sensors) or by changing it (actuators), they process the data and communicate to others. For performing their tasks Cooperating Objects can be equipped with other devices, e.g. storage.

The general idea of Cooperating Objects is to have several specialized and, thus, heterogeneous devices that can perform parts of a task in an efficient way. This modularization helps to keep the single devices simple and maintainable. The overall goal of a Cooperating Object is reached by cooperation of these devices. It seems clear that for this interaction with each other in a distributed environment, all of them need to be equipped with communication capabilities that can be based on wired or wireless technology.

The amount of effort devoted by a particular device to each task is determined on an individual basis. This is the main difference with respect to other related technologies such as passive RFID. In the case of Cooperating Objects, the intelligence of the system lies distributed in the network and each individual entity is able by design to perform complex processing tasks, if so needed. On the other hand, passive RFID does not perform any kind of processing and only returns an identification as an answer to an external stimulus (the reader). In this sense, the intelligence of a system based on passive RFID technologies lies in the infrastructure

and in the readers, but not on the distributed and embedded devices that form the bulk of the network. Active RFID tags feature their own power source e.g. battery or an external source, and are more advanced coming very close to the sensor domain.

In view of the emergence of new technologies and devices, their increasing integration into the everyday life and the need to coordinate them with a view to making communication easier mainly as to the interoperability, the mobility and the scalability, Cooperating Objects are regarded as a key enabler and aim at providing a proactive support to users or machines in their collaborative tasks. Indeed, the major advantage of the Cooperating Object lies in the possibility to tackle the complexity of the new surrounding environments due to the high number of involved devices or systems and the heterogeneity of components.

The generality of this model allows us to seamlessly include different fields like sensor networks, pervasive computing, embedded systems, etc. As an example, consider the following scenario: Nowadays, we have at our disposal lots of information, data sources or systems or even services, like the traffic panels along the main roads, traffic radio, GPS devices or web services to plan a trip. GPS devices may give alternatives but only under the human initiative and not all commercial GPS take into account real-time traffic data yet. However, video cameras and other surveillance systems are able to provide some of this data. And if we integrate the Traffic Message Channel services (TMC technology) for example, we get another useful flow of information which could be computed. To be able to achieve our goal, a pro-active process is necessary, and for this reason the co-operation between all of these various information sources is vital. How to reach that, i.e. a cooperative surrounding environment to link vehicle and infrastructures? One of the solutions would be to use Cooperating Objects, which would make it possible to drop some current barriers between these elements such as heterogeneity, complexity, scalability and to improve the communication with a view to providing ad hoc networks, thus data mobility would be enhanced. In this scenario, the Cooperating Object might consist of several parts: for instance one that continuously measures local traffic data, a second one to integrate all traffic-related data from available information flows from infrastructures and another that asks the GPS device in view to offering route alternatives.

2.1.1 Cyber-Physical Systems

During the last years the term “Cyber-Physical Systems” has been used, mainly in the US, to deal with systems similar to Cooperating Objects. However, no single and clear definition of this research area is available. In fact, several explanations can be found that set more or less narrow limits. We show two of them and compare them with Cooperating Objects.

In the NFS call 10515 [Nat09] a broad definition is given as follows:

“The term cyber-physical systems refers to the tight conjoining of and coordination between computational and physical resources. [...] These capabilities will be realized by deeply embedding computational intelligence, communication, control, and new mechanisms for sensing, actuation, and adaptation into physical systems with active and reconfigurable components.”

The Steering Group or the CPS Summit 2008 provided a more precise definition in its “Cyber-Physical Systems Executive Summary” [CPS08]:

“A CPS is a system:

- in which computation/information processing and physical processes are so tightly integrated that it is not possible to identify whether behavioral attributes are the result of computations (computer programs), physical laws, or both working together;
- where functionality and salient system characteristics are emerging through the interaction of physical and computational objects;
- in which computers, networks, devices and their environments in which they are embedded have interacting physical properties, consume resources, and contribute to the overall system behavior.”

While the first definition allows almost all Cooperating Objects to live within Cyber-Physical Systems as well, the second definition assumes the coalescence of physical and computational objects have to form a new type of system in which neither part can be distinguished.

Important in both definitions is the tight integration of physical and computational objects although the degree of integration is different. Cooperating Objects are also interacting with the physical world but mostly they still can be seen as separate systems with a clear interface to the environment with specific characteristics. They also focus on the composition from their single parts by cooperation as stressed by the definition given above.

2.2 Research Areas

Nowadays, it is impossible to create or work on technologies that do not rely more or less heavily on the development of other areas. New developments in these related areas usually go hand-in-hand, and a major breakthrough in one of the enabling technologies can really boost the work that can be performed on the other areas.

This is also true for Cooperating Objects and, as we have seen in the previous sections, Cooperating Objects have emerged as a combination and natural extension of already existing research areas that have been evolving rapidly in the past years.

Therefore, it is worth pointing out more precisely what we consider are the major pillars for research in Cooperating Objects since this classification is also used in several chapters of this roadmap.

Hardware: Obviously, all Cooperating Objects need some hardware platform to be built on. Although some standard platforms exist for many Cooperating Objects areas, e.g. MicaZ or TelosB in Wireless Sensor Networks, specialized platforms with new constraints are used for single scenarios. In general, two development directions are taken due to advances in electronics: either the current limitations of the platforms concerning speed or memory are expanded while keeping external requirements to energy or space, or the form factors and energy requirements are decreased while keeping the capabilities of the platforms. Since power is an essential problem of many Cooperating Objects, new ways of saving or harvesting energy are explored.

Sensors and actuators are important as well due to the tight interaction of Cooperating Objects with the environment. Beside the two general platform research directions, the usage of several cheap sensors instead of a single expensive one or the (re)calibration of sensors is a challenging task.

Algorithms: Several standard tasks occur in almost every Cooperating Object and research community is tackling them in many ways that fit more or less well to Cooperating Objects. Starting with communication we can name the management of radio resources (frequencies or time), MAC and routing protocols or bandwidth estimation. Since communication is essential for cooperation, improvements in this area largely affect the behavior of the applications. To be able to react based on sensed information, data has to be processed, stored

and queried inside the network. Depending on the type of data and the characteristics of the network this can be very challenging. To correlate data from different sources, it is necessary to have a common time base, which can be achieved by time synchronization, and a good knowledge of the position of the data source, which is the responsibility of localization algorithms. Temporal and spatial coordination is also a fundamental issue in the coordination and cooperation of mobile Cooperating Objects and, particularly, in the coordinated control and cooperation of autonomous mobile robots with on-board navigation and decisional capabilities [Cue05].

Although more algorithmic areas could be identified for the overall domain of Cooperating Objects we have identified only those that have a greater impact on the whole research field.

Non-functional Properties: Properties of a system that affect its quality but not its functionality are called non-functional properties. Compared to traditional Quality of Service criteria the perspective in Cooperating Objects needs to be extended to include unique features of this research area. The cooperation between several specialized and, therefore, different devices implies the property Heterogeneity and due to their possibly vast number scalability is of great importance. Timeliness with respect to throughput, delay and real-time stems from the embedded systems sub-domain with short-cycled control loops. Due to the integration of Cooperating Objects in the real-world systems need to be reliable and robust. Security and privacy issues arise from the fact that events in the real-world are monitored that can expose private data and that this information has influence on the behavior of the system. Therefore, Cooperating Objects have to deal with tampering attempts. Finally, Cooperating Objects can be mobile which is both the controlled behavior of a system and a property that the system has to deal with.

Systems: To ease the development of software for Cooperating Objects Operating Systems provide basic functionality, e.g. hardware access or memory management, that the algorithms can rely on. There also exists programming abstractions and middleware approaches to further hide the complexity of the platforms or to provide additional functionality compared to the operating system, which can also be special for a certain application area. A special case of middleware deals with the integration of Cooperating Objects into larger systems, for example business applications, or the combination of several Cooperating Objects. To be able to debug and manage already installed Cooperating Objects special functionality has to be included on the devices itself and tools have to be provided for the owner to deal with the gathered debugging information.

Others: This category does not affect the software of Cooperating Objects itself but helps in their development and installation. Standardization of hardware and software is necessary to enable the interoperability and, thus, the cooperation of many Cooperating Objects. Otherwise all solutions will remain a separate system and will not profit from each other. Standard components can be evaluated thoroughly in simulators and testbeds to find errors before actual deployment. Finally, modeling and planing tools help to design the deployment of the system.

The Emerging Domain of Cooperating Objects

Marron, P.J.; Karnouskos, S.; Minder, D.; Ollero, A. (Eds.)

2011, XXVII, 271 p., Hardcover

ISBN: 978-3-642-16945-8