INTERNATIONAL DESIGN CONFERENCE - DESIGN 2010
Dubrovnik - Croatia, May 17 - 20, 2010.

DESIGN 2010

COGNITIVE PRODUCTS: DEFINITION AND FRAMEWORK

T. Metzler and K. Shea

Keywords: cognitive products, cognitive technical systems, cognitive capabilities, mechatronics, new product development

1. Introduction

Many products in our everyday life today are mechatronic products that integrate mechanical, electronic and software components. To develop this type of product an interdisciplinary approach combining mechanical engineering, electrical engineering and computer science is necessary to exploit their complete potential. Mechatronic products are multidisciplinary, complex systems that capitalize on the synergies of these disciplines to enable new product solutions that exceed the capabilities of today's products. There are several products on the market that call themselves adaptive, smart, intelligent or similar. Equally, these terms are used widely in a research context related to products. However, no common definition of what these products are and what properties they possess exists. This paper gives an overview of definitions of what these products are, what differentiates them and what will be the next generation of cognitive products.

Achieving product properties like robustness, reliability, flexibility and autonomy are central to product development. One approach is to make products more intelligent in order to handle tasks autonomously, react robustly to surprise in new environments and situations and provide a high level of user interaction and adaptivity to different users. Current research provides new methods and indicates that cognitive technical systems (CTSs) are a key to successfully developing such products. Cognitive products offer great potential to satisfy user needs and desires through cognitive capabilities but also make high demands on the development process.

This paper begins by reviewing existing products and systems related to cognitive products. Based on this review, a definition of cognitive products is developed, highlighting the capabilities of cognitive products and their differences to existing products. The definition of cognitive products in turn is used as the basis for a framework for cognitive products. This framework is presented and illustrated with a range of cognitive products developed in student projects over the last three years.

2. Background

Previously, products have been classified by their components according to the field of engineering they belong to, e.g. mechanical-, electronical- and software engineering. Nowadays products are commonly differentiated by their capabilities and not solely by their components. Even though some products use the same hardware components, their capabilities can vary tremendously. According to their capabilities, products can be classified as mechatronic, flexible, adaptable, smart, intelligent or even cognitive due to increasing "intelligence" stemming from embedded software and hardware. To date, there is no universal way to classify these products according to their capabilities. The products are all linked to one another and the transition from one to another is fuzzy and strongly depends on individual definitions and terminology. The following background chapter gives an overview of definitions and characteristics of related product types in order to define and distinguish cognitive products.

2.1 Mechatronic Systems

Mechatronics is a collective term for systems combining mechanical, electrical and software components. Mechatronic systems are linked between the fields of mechanical engineering, electrical engineering and computer science and use synergies of all three disciplines. A key characteristic of meachatronic systems is the functional integration of sensors, actuators and data processing. The result is a multidisciplinary complex system. Figure 1 shows a general scheme for a mechatronic system. One of the goals of mechatronics is to improve the behavior of technical systems by collecting information from the environment and the system itself using sensors. This information is processed to determine, depending on the situation, an appropriate or optimal reaction using the systems actuators. Mechatronic products are able to detect changes in their environment and react accordingly using control algorithms [VDI 2206 2004].
The benefit of mechatronic products is that through the cooperation of different disciplines new solutions become possible and technically feasible. Mechatronic systems can improve the cost-benefit ratio of existing products and lead to new product ideas. They play an important role in the automation of product functions.

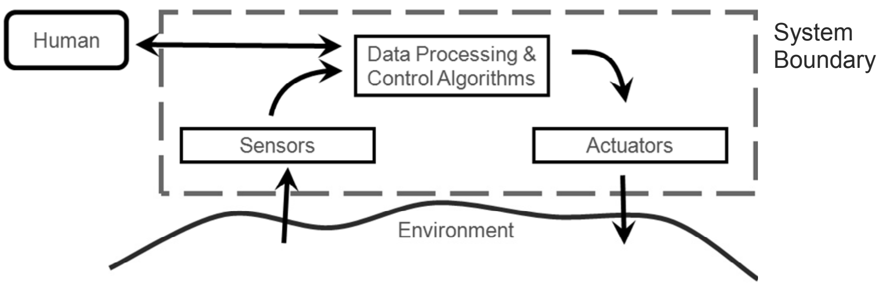


Figure 1. General Scheme for a Mechatronic System [according VDI 2206]

2.2 Adaptable Products

Similar to mechatronic systems, there are various definitions for adaptable products in literature but no definition suits all requirements. Mechatronic systems that accommodate predictable changes are called adaptable [Chmarra et al. 2008]. In general there are three types of adaptability related to different phases of the lifecycle: design time adaptability, runtime adaptability and lifetime adaptability. As implied by the names, the adaptation of a product can take place during the design time or during runtime when the product performs a task. Another possibility is the adaptation of a product’s lifetime by prolonging the service life in its normal operational mode and by adapting it to new operational modes. Furthermore, adaptable products can be classified by how the adaptation takes place. One possibility is that the product adapts to new requirements automatically while another possibility is that a user carries out the adaptation. Further, the adaptation can be done in real time or, when the system is not in use [Chmarra et al. 2008].
In this paper, automatic runtime adaptability is considered the category of adaptable products most related to cognitive products. According to Chmarra et al. [2008], adaptability is defined as system parameters that can be changed to improve performance of the system in predictable situations. Only the system can change the parameters and this is done autonomously. The goal of adaptable products is to extend the utility of a product and its design. Adaptable products are able to match a wide range of working conditions caused by user requirement changes or by environmental changes.
To date, no standard theoretical framework for designing adaptable products is established. Only a few rules, generalized methods, software tools and guidelines for Design for Adaptability (DfA) are available. Chmarra proposes a general scheme with seven components of an adaptable product, as shown in Figure 2. The two ellipses in the scheme show the main components of an adaptable product.

They are responsible for taking actions according to the input information. The other components, depicted in rectangles, are inputs and outputs of the two main components.

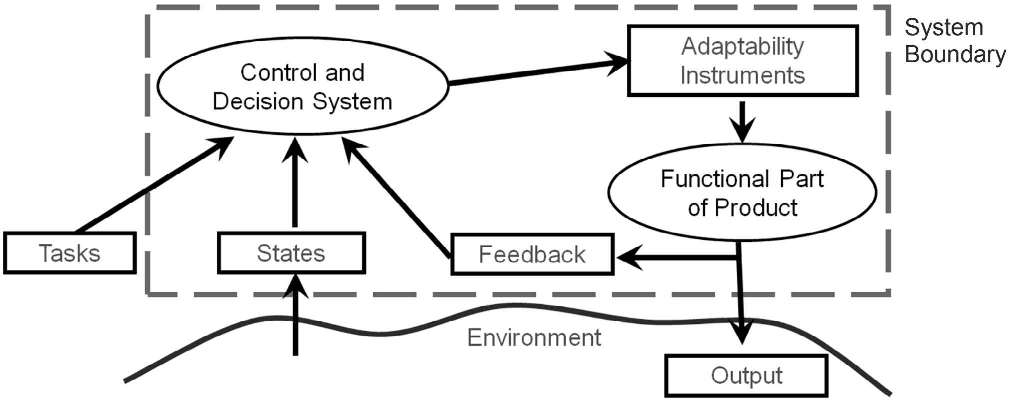


Figure 2. Scheme for an Adaptable Product [according to Chmarra et al. 2008]

Tasks represent an input from the user to the control and decision system. Usually tasks are user demands. States provide input from the product itself to the control and decision system. Compared to mechatronic systems, tasks correspond to a human intervention and states correspond to sensor patterns. States are configured using sensors that are collecting internal data of the product as well as data of the outer environment. If the input of an adaptable product changes, the control and decision system is able to recognize it and determine the appropriate action. To do so, an adaptable product needs a set of predetermined “action plans” to adapt to the changing environment. The control and decision system corresponds with the data processing and control algorithms of a mechatronic system. Adaptability instruments are a list of possible parameters and actions that are available to the control and decision system in order to adapt the product. They determine to what degree a product can be modified. The modification is finally accomplished by the functional part of the product that executes the work to be done according to the actuators in a mechatronic system. This results in an output. The internal feedback, gained by sensory information of the output, is linked back to the control and decision system and closes the control loop [Chmarra et al. 2008].

A key point is that adaptable products are only able to adapt to a changed environment if the designer has foreseen the changes that can occur and derives appropriate action plans. These action plans are carried out using a combination of adaptability instruments.

2.3 Smart Products

Smart products have several abilities that differentiate them from adaptable products. The architecture of smart products is illustrated in Figure 3. They can perceive the environment using sensors inside or outside the product as well as receiving environment data from external devices or sources. For example, a smart phone can take pictures using the integrated camera but also download pictures from the internet or another external source. Smart products process the sensory data and support the user dealing with an increasing complexity of everyday life. Responsible for the “intelligence” of smart products is access to data and knowledge and the interaction with it through the central processing unit. The combination of knowledge and perception in terms of sensory data enables smart products to know their current state. They are context aware up to a certain degree, determined by the quantity and quality of sensors. Knowledge of their current state and context awareness allows these products to interact, network and communicate with their environment including humans, robots or other products in a way that can be called “smart”. Smart products are able to handle tasks with a more demanding functionality compared to adaptable products [Acatech 2009].

However, smart products can neither reason about aberrations between sensory data and knowledge models nor adapt to these aberrations by creating new algorithms. They execute the preprogrammed algorithms and rely on the information stored in the knowledge models. Because they do not have actuators they can not adapt to situations that require a physical adjustment of the product.

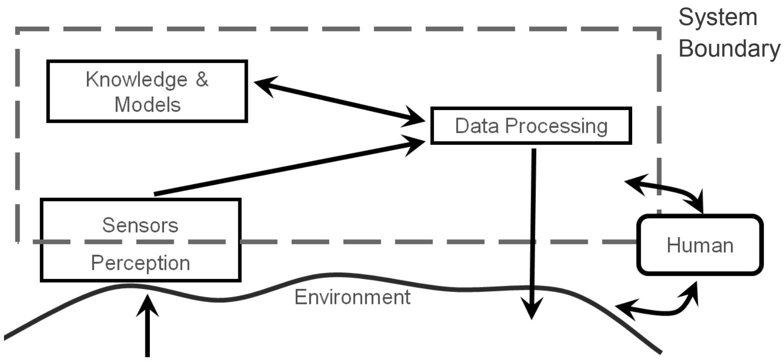


Figure 3. Smart Product Architecture

The minimum hardware requirements of a smart product are at least one processor, memory and software. Additionally sensors, network connection, energy supply, RFID transponders, multiple processors, memory, etc. can be integrated if the circumstances require it. The miniaturization of components made it possible that the technology behind these products can be easily embedded and became nearly invisible [Kranz 2005].

Smart objects are changing the character of workflow as well as the organization of private life, since they can be found everywhere. A classification of these products can be done considering the distance to the human. The products can be implanted, integrated in clothing, it can be an object to carry around, a mobile infrastructure or even an immobile sensor network in a building.

2.4 Cognitive Technical Systems

Cognitive systems, no matter if human or machine, collect data of their environment and their own status through sensors. This data is processed and interpreted in order to generate an appropriate response through actuators. Key to cognition is that all data is processed and interpreted according to the perceived situation and not according to a rigid control algorithm. From the cognitive psychology point of view, the human is the most cognitive system considering complexity and capabilities. The focus of research around cognitive technical systems (CTS) is to achieve similar levels of flexibility, adaptivity, and robustness as found in humans. This can only be achieved by implementing cognitive capabilities in technical systems, for example, to perceive, learn, reason and plan. But cognition does not only emerge from the single components but rather through the interaction of different cognitive capabilities. Compared to mechatronic systems mentioned in previous sections, additional research areas are involved in the development of CTSs, for example psychology and neuroscience that help to better understand the working principles of natural systems and develop appropriate mappings of this understanding to technical systems.

To date, the basis of many technical systems, like those presented before, is a reactive structure, performing always the same output for a given input. Any mechatronic system according to the VDI 2206 using a closed control loop has a reactive structure. Adaptable products can only adapt to predictable situations and smart products can react only according to the knowledge they are able to access but do not have actuators to adapt themselves to changed conditions. In contrast, CTSs do not necessarily perform the same output for a single input; the system is able to adapt to predictable and unpredictable conditions using actuators. Thus, the situations and responses of CTSs are not predetermined like in reactive structures. In contrast to mechatronic products and systems, CTSs have adaptable and flexible control loops [Beetz et al. 2007].

Brooks claims that problem solving behavior, language, expert knowledge and its application, and reasoning are simple tasks once the “essence of being and reacting” are available. The “essence of being and reacting” are the ability to move around in a dynamic environment and sensing it [Brooks 1991]. This emphasizes the importance of CTSs that they do not only use a predetermined behavior. Giving CTSs a physical grounding makes them “truly” intelligent systems able to perform actions in a changing environment. According to Brachman CTSs are “systems that know what they are doing”

[Brachman 2002]. Derived from this statement, CTSs further must be able to explain what they are doing and why. Therefore, they must be actively integrated in the environment and have the ability to exchange information. CTSs are flexible, possess an environment-adaptive task-control and have the ability to reflect relevant data of the environment. Furthermore, they need the ability to learn and anticipate [Paetzold 2008]. This makes adaptability an important property of CTSs. They have the intention of improving their performance and to reduce waste in resource utilization. Further properties of CTSs are a high level of interaction with humans, integration in a living environment and to act "human-like" [Kinsner 2006].

Derived from all these observable capabilities, basic cognitive capabilities can be identified. However, in literature there is no accurate definition about which, and how many cognitive capabilities a system needs, to be characterized as a CTS. The following list summarizes the cognitive capabilities of a CTS with high performance and tending to "human-like" performance: *to perceive, to learn, to plan, to reason from knowledge models, to have a model of itself, to be environment aware, to communicate, to interact, to perform actions in unstructured environments* [Brachman 2002; Shea et al. 2008] *and to use natural language* [Paetzold 2006]. The meaning of these cognitive capabilities is explained in the next section about the architecture of CTSs.

Several enablers have allowed the recent development of CTSs. Artificial Intelligence (AI) and related disciplines such as machine learning and speech processing have improved significantly in recent years. The growing processing power makes real time processing possible and neural science contributes important insights about the workings of the human brain. This creates great opportunities to capitalize on the growing knowledge and methods for CTSs and explore how cognitive capabilities can be embedded in consumer products.

2.5 Architecture of Cognitive Technical Systems

An architecture for CTSs is now presented, according to Beetz et al. [2007] and describes the organization and structuring of CTSs that can be used to design and realize new systems with cognitive capabilities. The architecture is a guideline describing the required components and how they are linked in order to implement cognitive capabilities in a technical system. In other words, the architecture of a CTS explains the inner workings of the system.

There are several properties a CTS has to be able to accomplish according to the previous section and therefore have to be considered in the architecture. With the aid of Figure 4, these properties are now briefly described. At first CTSs have to be able to perceive their environment using sensors so that the system becomes environment aware and is able to develop a model of itself. Perception is the basis for most other cognitive capabilities.

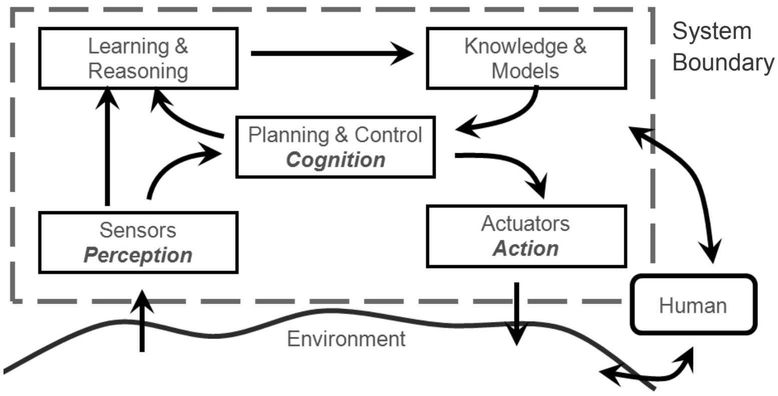


Figure 4. Cognitive System Architecture [according to Beetz et al. 2007]

Learning, or more precisely machine learning, allows machines to learn models and change their behavior based on the perceived sensor data. Learning can be achieved with different algorithm types depending on the available input and desired outcome. Most common algorithms can be divided into supervised and unsupervised learning. In general, CTSs can learn in two ways: from experience and by being instructed [Brachman 2002]. Non-cognitive systems always react in the same way according

to one stimulus; in a CTS, however, this is different, the reaction can be changed through learning. Paetzold describes this as the main difference between cognitive and non-cognitive systems [Paetzold 2008]. Further, CTSs are able to reason. This implies that they are reflective enough to recognize when heading down a blind alley or ask for information if it is not available by further reasoning [Brachman 2002]. Knowledge and models are not only stored in a static memory but information can autonomously be added or changed through learning and reasoning algorithms.
The core of a CTS is the “cognitive” unit performing planning and control tasks. Planning is done in order to achieve single or multiple goals. The necessary information to create a plan is supplied by perception through sensors and internal knowledge models, which can be provided or learned. Usually a set of actions is required to achieve one goal. CTSs are able to anticipate the actions to achieve a goal, in other words they can create a plan. After creating a plan, control algorithms transform a plan into actions that are carried out in the physical world. Therefore CTSs need actuators that enable them to perform actions. Further, CTSs need the ability to communicate and interact with their environment and especially with humans. By combining of the above mentioned capabilities CTSs are able to explain what they are doing and why [Brachman 2002]. The capabilities and architecture for CTSs will now be explored within the context of developing cognitive products.

3. Cognitive Products: Definition and Framework

To date, cognitive products have not been investigated in the scientific literature. Most of the discussion focuses on CTSs and at most touches on robots as products. Nevertheless, terms like “intelligent”, “smart”, “adaptable”, “cognition” or “cognitive” are present in the media and became buzzwords declaring new products as smart, etc. This chapter gives a definition of cognitive ,products to identify their abilities and characteristics. The definition helps to differentiate different types of products and further will help to develop them. A framework for cognitive products is then presented.

3.1 Products

From the product development point of view, products are devices that provide a service that enhances human experience. They succeed when they play a major role in creating optimal experience for customers [Cagan and Vogel 2007]. In marketing, products are described as anything that can be offered to a market and is able to satisfy a want or need. Regarding this definition, products further can be tangible or intangible, durable or non-durable [Kotler et al 2007]. In this paper, product is used for a tangible and durable thing, created through labor and able to satisfy a need when offered to a market. Following the definition above, a crucial point of cognitive products is to satisfy customer needs. This is what differentiates products from systems.
Jones describes several principles that have arisen during the development of the iRobot Roomba and likewise apply to the development of cognitive products. It is pointed out that consumers do not want a robot or product that can do “anything” if programmed. Rather, they want a robot or product that meets a defined consumer need that they can identify with [Jones 2006]. Essential is that the product has functions that are useful, usable and desirable for potential customers [Cagan and Vogel 2007]. Developers of cognitive products have to concentrate on how to match emerging cognitive capabilities with user needs and desires since this is what matters most from customers point of view. Every new product, cognitive or not, has to compete with products and services on the market fulfilling the same or a similar task. In fact, it has not only to fulfil the task competitively but also the cost has to be comparable. Instead of adding gadgets, additional functionality should improve the product in the eyes of the user or make the product more reliable. Another reason to exclude nonessential systems is to minimize complexity. Customers only buy new products if they benefit from them [Jones, J. L. 2006]. Designers should always consider these principles during the development process of cognitive products.

3.2 What are Cognitive Products?

Fusing the definitions for products and CTSs a definition for cognitive products is now proposed:
Cognitive products are tangible and durable things with cognitive capabilities that consist of a physical carrier system with embodied mechanics, electronics, microprocessors and software. The

surplus value is created through cognitive capabilities enabled by flexible control loops and cognitive algorithms. Customer needs are satisfied through the intelligent, flexible and robust behavior of cognitive products that meet and exceed customer expectations. Cognitive products have all or a subset of capabilities of CTSs and the solid grounding of an everyday product that meets user needs and desires.

What makes products cognitive are their special properties stemming from the integration of CTSs. The implementation of cognitive capabilities results in high-level capabilities, for example:

- Cognitive products act not only autonomously but in an increasingly intelligent and human-like manner, as opposed to deterministic control of machines and mechatronic systems,
- Cognitive products adapt to a dynamic environment robustly,
- Cognitive products can be integrated into human living environments and have a high level of interaction and cooperation with humans,
- Cognitive capabilities result in products with higher reliability, flexibility, adaptivity, interaction and improved performance,
- Cognitive products should be able to maintain multiple goals and make appropriate decisions.

To achieve these high level capabilities, more specific cognitive functions are needed and are introduced in the following section.

3.3 Framework for Cognitive Products

In this chapter a framework for cognitive products is presented, based on the definition of cognitive products and their properties. To provide a better understanding of the framework, it is supplemented with five example products with cognitive capabilities. Table 1 introduces five products with cognitive capabilities that have been developed within a student, project-based learning class. Teams of three to five students from different engineering disciplines develop cognitive products within target scenarios. Target scenarios have included developing a cognitive oven for the “intelligent” kitchen and developing cognitive toys for young and old. The students chose the most promising product idea generated, create a concept and build form and functional prototypes. The seminar is supported by focused lectures on cognitive methods as well as product development methods and an introduction to prototyping [Shea et al. 2008].

In general, concepts of cognitive products are developed by the students and the primary functionalities are integrated in the functional prototype. To enhance the product concepts and prototypes further towards cognitive products, some projects are continued after the seminar.

Table 1. Products with Cognitive Capabilities Developed within Student Projects

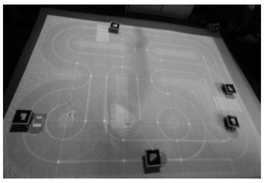
coffee waiter	cognitive oven	robot swarm game	Carrera slot car	linked sockets
				

Table 1 shows five pictures of products developed. A short description of each is now given. The coffee waiter serves coffee in a learned environment to several users. For ordering coffee, the user communicates with the coffee waiter via the Internet. Depending on the requests for coffee and internal resources, it calculates an efficient way to serve the users and autonomously navigates within the environment. The cognitive oven prepares meals according to a learned user profile and improves the cooking program and results through feedback from the user. The robot swarm game platform consists of five robots moving autonomously on a virtual map. One robot is driven by a user and interacts with the other robots on a displayed map. The product can be used to teach children, for example, traffic rules and adapt autonomously to their skills. A virtual opponent for a Carrera slot car game is great fun if you want to play but do not have a human opponent. The virtual opponent learns a model of your driving for your fastest lap time and adapts to it. During a race, the opponent drives tactically, e.g. staying just behind the user, to keep the game fun. The last example is a product to save energy in the household. The user plugs networked sockets between different devices and the power supply. The product learns the daily schedule of use of products and can turn electronic devices off or from standby to off when a product is not likely to be used.

Table 2. Framework for Cognitive Products

cognitive functions		coffee waiter	cognitive oven	robot swarm game	carrera slot car	linked sockets
learn	locations	☒		☒		☐
	point of time	☒	☒			☒
	behavior model		☒		☒	
	environment model	☒				
	use of resources	☒	☒			☒
	user models		☒	☐		☒
know	environment model	☒		☒	☒	☒
	user model	☐	☒	☐		☒
	model of itself	☒	☒	☒		
reason	task fulfillment					
	alternative actions					
plan	route	☒		☒		
	schedule	☒	☒	☒		☒
	use of resources	☒	☒			☒
perceive	location	☒		☒	☒	
	user recognition		☒	☐		☒
	object recognition	☐	☒	☒		☐
communicate & interact	user adaption	☐	☒	☐	☒	☒
	teach user			☒		
	give feedback	☒	☒			
act	autonomous movment	☒		☒	☒	
⋮	⋮	⋮	⋮	⋮	⋮	⋮

☐ = Part of the Concept ☒ = Implemented in the Prototype

In Table 2, cognitive capabilities of the cognitive products shown in Table 1 are listed. The general cognitive capabilities, expressed by a verb, are supplemented by nouns, expressing a specific characteristic, and become cognitive functions. These type of functions are commonly used in product development methods. The framework does not claim to be exhaustive in all points but covers the cognitive capabilities proposed for CTSs in section 3.2 and introduces cognitive functions.

Cognitive functions of the framework are now explained through the coffee waiter example. The coffee waiter has the capability to learn several things. It learns the map of the environment by having a human steer the robot through the environment. It records obstacles using a laser range scanner and autonomously creates a model of the environment based on the sensor data. Using this model, it can perceive its location within the learned map. Likewise, the robot learns locations of interest, for example the kitchen is important to refill coffee. Compared to mechatronic products, e.g. the Roomba vacuum cleaner, this capability enables the robot to navigate precisely through it’s environment. Further the robot can learn if locations change and adapt to the new environment. Learning points of time enables the coffee waiter to know when a specific user usually orders coffee or when a peak demand is likely to emerge. Learning the use of resources enables the coffee waiter to know several parameters about itself. It can learn that the coffee carafe is big enough to carry coffee for five cups according to input data of force sensors and how much is usually taken by one user. After this learning process, it knows how much coffee it can carry in total, how much is currently left according to the residual weight and how many users can still be served. It also learns and knows the characteristics of the battery capacity. If the coffee waiter gets more than one order through the web interface it autonomously creates a plan of how to serve the users efficiently. Therefore, an online travelling salesman algorithm is used that recalculates the route everytime a new order arrives. Further, the robot can plan the use of its resources, e.g. battery and coffee and schedule trips to the kitchen to refill coffee or charge the batteries. If no orders are waiting, the robot charges its batteries and refills coffee. Only if it anticipates that most likely no more coffee is ordered, e.g. at weekends, it adapts to the new situation and stays at the docking station. To date, the coffee waiter can only perceive objects in its way when it hits them but can not recognize what it hit. In the future, an obstacle avoidance is planed to recognize static and dynamic objects and drive around them. An obstacle avoidance makes the coffee waiter much more flexible and it can plan a different route to reach its target. More interaction between the robot and the user is also envisioned. The robot can interact with users by giving them feedback, e.g. the current coffee level is displayed on the web interface or through a display mounted on the robot asking for the coffee carafe if the user does not return it. How long the coffee waiter will wait before it moves on to the next target depends on the single user and is adapted individually. The sum of all the cognitive capabilities described enable the coffee waiter to move autonomously, know about its own state and the environment, plan, schedule, interact with users and in general "know what it is doing". In the future, further cognitive functions will be integrated, especially the ability to reason about the further coffee consumption depending at the time of the day.

To summarize, several cognitive products are developed to explore how the ideas, methods and architecture of CTSs can be used to create a new generation of products. The framework in Table 1 makes clear that there is no product created that possess all cognitive functions but rather that different products exhibit different functions that are embedded to meet user needs and create new experiences for users. So far, only the ability to reason has not been investigated and is left for future work. Together, the products developed provide a proof-of-concept that cognitive functions can be integrated into products to enhance their functionality.

4. Discussion

Cognitive products are the logical evolution of mechatronic and adaptable products enabled through mapping customer needs to emerging cognitive capabilities. Additional value is not only created through new hardware components but especially through their interaction and interaction with new software capabilities. By adding specific characteristics to the general cognitive capabilities they become cognitive functions that are technically feasible. Like functions commonly known in product development, cognitive functions are described by a verb and a noun with the verb expressing the cognitive capability and the noun supplementing it. These cognitive functions can be integrated in products but so far no methods support this approach. The next step is to build tools that support the systematic development of cognitive products methodically. A design catalog could be used to collect already known solutions of cognitive functions and serve as a knowledge base for future applications.

Before cognitive products can be offered to the market, testing is a very important issue. Only if a product can prove its abilities in the real world, it will satisfy customers. This is especially important for cognitive products since they should respond robustly in new and unforeseen environements and situations. Further, the response of the product is no longer deterministic but situation dependent. To master this challenge, real world testing is necessary and cannot be replaced by simulation completely. Appropriate methods for testing cognitive products have to be developed.

The present definition and framework for cognitive products will be further extended and introduced in the student seminars to guide the students developing cognitive products. Further development of the existing products presented to move further towards achieving more complete cognitive functionality is also underway.

5. Conclusion

This paper presents a definition of cognitive products derived from reviewing CTSs and other mechatronic products and systems. They are described by their properties and how they differ from mechatronic, adaptable and smart products. Cognitive products are the next generation of products. However, there is neither one available on the market nor are there any guidelines of how to develop them. The definition and framework proposed in this paper are the first steps towards defining and developing cognitive products. Using products developed within a project based student seminar, examples for the realization of cognitive capabilities are provided. Further research in the field of cognitive products is necessary to systematically create successful cognitive products.

Acknowledgement

This research is part of the Innovation@CoTeSys project within the Cluster of Excellence, Cognition for Technical Systems – CoTeSys (www.cotesys.org), funded by the Deutsche Forschungsgemeinschaft (DFG). The authors want to thank all participants and coaches of the Innovation@CoTeSys seminar and in particular Bergen Helms and Andreas Kain for their discussions about important aspects of this paper.

References

Acatech DISKUTIERT, "Intelligente Objekte, Technische gestaltung – Wirtschaftliche Verwertung – Gesellschaftliche Wirkung", O. Herzog, T. Schildhauer (Eds.), Bremen, 2009.

Beetz, M. et al., "Cognitive Technical Systems – What is the Role of Artificial Inteligence?"KI 2007: Advances in Artificial Intelligence, Vol. 4667, Berlin, Springer 2007, pp. 19-42.

Brachman, R. J., "Systems That Know What They're Doing", Intelligent Systems, Vol. 17, No 6, 2002, pp 67-71.

Brooks, R. A., "Intelligence without representation", Artificial Intelligence Vol. 47, 1991, pp. 139-159.

Cagan, J., Vogel, C. M., "Creating Breakthrough Products - Innovation from Product Planning to Program Approval", FT Press, 2007.

Chmarra, M. K., Arts, L., Tomiyama, T., "Towards Adaptable Architechture", Proceedings of the ASME 2008 International Design Engineering Technical Conference & Computers and Information in Engineering Conference IDETC/CIE 2008, pp 367-376.

Jones, J. L., "Robots at the Tipping Point. The Road to the iRobot Roomba, Robotics & Automation Magazine, IEEE, Vol. 13, Issue 1, 2006, pp. 76-78.

Kinsner, W., "Towards Cognitive Machines: Multiscale Measures and Analysis", Proceedings of the 5th IEEE International Conference on Cognitive Informatics, Yao, Shi,Wang and Kisner (Eds.), Beijing, 2006, pp 8-14.

Kotler, P., Keller, K. L., Bliemel, F., "Marketing-Management", Pearson Studium, München, 2007.

Kranz, M., Schmidt, A., "Prototyping Smart Objects for Ubiquitous Computing", Proceedings of the International Workshop on Smart Object Systems, 7th Int. Conference on Ubiquitous Computing, Tokyo 2005.

Paetzold, K., "On the Importance of a Functional Description for the Development of Cognitive Technical Systems" Proceedings of the 9th International Design Conference – DESIGN 2006, D. Marjanovic (Ed.), FMENA, Zagreb, 2006, pp. 967-974.

Paetzold, K., Schmid, U., "To the Development of a System Architechture of Cognitive Technical Systems", ", Proceedings of the 10th International Design Conference – DESIGN 2008, D. Marjanovic (Ed.), FMENA, Zagreb, 2008, pp.1311-1318.

Shea, K., Engelhard, M., Helms, B., Merz, M., "Teaching an Integrated new Product Development Seminar on Cognitive Products", Proceedings of the 10th International Design Conference – DESIGN 2008, D. Marjanovic (Ed.), FMENA, Zagreb, 2008, pp. 1401-1408.

VDI-Gesellschaft Entwicklung Konstruktion Vertrieb, "Design methodology for mechatronic systems", VDI 2206, VDI-Gesellschaft Entwicklung Konstruktion Vertrieb, 2004.

Torsten Metzler
Scientific Assistant
Virtual Product Development, Product Development, Technische Universität München (TUM)
Boltzmannstr. 15, 85748 Garching, Germany
Telephone: +49-89-289-15154
Email: torsten.metzler@pe.mw.tum.de
URL: http://www.pe.mw.tum.de