Semantic Intelligent Space for Ambient Assisted Living

István Fülöp, Péter Galambos, Péter Baranyi

Semantic Intelligent Space for Ambient Assisted Living

István Marcell Fülöp, Péter Galambos, and Péter Baranyi

Computer and Automation Research Institute,
Hungarian Academy of Sciences, Budapest, Hungary
{ifulop,galambos,baranyi}@sztaki.mta.hu
http://www.coginfo.sztaki.hu

Abstract. Today, the research field of assisted living technologies is becoming more and more important. In this paper, an intelligent space framework is going to be presented which can be applied as a framework for assisted living. There are two beneficial properties which can improve the performance of the presented framework. The first is the use of semantic information which increases the flexibility and the autonomy of the framework so that it maximizes the collaboration possibilities between the entities of the intelligent space. The second is the relation to cognitive info-communications which makes it possible to use not only conventional channels in user interaction, but other alternative channels as well, i.e. to represent the user as a complex source and drain of several kinds of information. In this paper, it is described how these properties of the framework can be adopted in this special application field.

Keywords: Intelligent space, ambient assisted living, semantic information, ontology, cognitive infocommunications, future Internet

1 Introduction

Today, in the time of ageing societies, assisted living technologies are becoming more and more important fields of research. The Ambient Assisted Living Joint Programme of the EU [1] supports half a hundred projects dealing with different aspects of AAL. These projects can be sorted in three main categories regarding the objectives: Prevention and management of chronic condition of elderly people (e.g. [2]); Social interaction of elderly people (e.g. [3]); Independence of elderly and disabled people (e.g. [4]). Advanced technologies e.g. robotics, infocommunications, cogntive sciences or language technologies are all incorporated under the umbrella of AAL research. These technologies render really impressive results separately, however the real breakthrough seems to be far away. The high-level control of the devices is an advanced domain of AAL research. In order to realize high-level control, high-level autonomy at all levels of intelligence in devices is required as well as ergonomic methods for information-rich communication in efficient user interaction. In this paper, it is assumed that a general intelligent space framework can be fit to the special requirements of an ambient assisted living environment. In section 2, it is going to be presented, what kind of framework is needed and chosen for our purposes. Then in section 3 we present, what kinds of special requirements arise and how the general framework is extended in order to match these requirements. Section 4 describes, how the individual parts of the framework are realized, and how the different parts are related to each other. Section 5 gives simple application examples in order to show the capabilities of the framework. As a conclusion, in section 6 the capabilities of the applied technologies are evaluated.

2 Platform for connection

2.1 Physical connection

To provide a framework for the management of the intelligent space, where operations can be decomposed into configurations for the devices, the components of the framework, e.g. the wrapper components for the devices, the operation manager component, the component for user requests need to be connected together physically. The connection has to provide potential for the realization of the protocols between the components. The single components need to be added to and removed from configurations dynamically, in run time, e.g. if a wrapper component for a device is added or removed. The configurations of the components should be monitored and edited easily through some editor tool.

For these purposes, the RT-Middleware system seems to be a good solution. [5] The RT-Middleware framework contains components, which are separate executable programs, running on computers connected to the internet. The components have their own data and service ports, which can be dynamically connected together with some editor tool. If the components of the intelligent space are considered, every component needs to have a representation of the space itself, i.e. the room, where the devices are located together with the user. It seems to be a good idea to create one common representation of the common space, and share this representation between the components. The common representation should contain all of the relevant information, e.g. the location of the single devices, the location of the user, other content of the space, which is available for all of the components.

For this goal, the VirCA platform based on RT-Middleware seems to be a better solution than RT-Middleware itself. [6] The main feature of VirCA is virtual or real space management. The VirCA platform is based on RT-Middleware, VirCA components are RT-Middleware components as well, but the configuration of the components is different. In VirCA, there are three basic types of components. The first type is the virtual space itself, which provides space management, physics simulation, virtual objects and connection for cyber devices as real objects in the virtual space. The second type is cyber device, which is a wrapper for some real or simulated device, which can be placed in the virtual space in order to manipulate its content. The third type is the type of special components, e.g. input components for user input. The virtual space component handles basic user input as well as user output, i.e. the visualization of the virtual space.

In VirCA, different devices are not connected directly to the operation manager, but through the virtual space. The devices wrapped up into cyber devices together with the operation manager component are connected to the space manager component. Through the virtual space, components can send massages to each other. So, while cyber devices can dynamically be added to and removed from the virtual space, the devices can be connected to the operation manager through the services of the platform.

2.2 Logical connection

In the intelligent space for assisted living framework, the logical configuration of the components differs from the physical. The low-level physical configuration is intelligent space oriented. This is based on the concept of VirCA, where individual cyber devices can join a common virtual space in order to realize some collaboration. [7] The devices can collaborate in the sense that they are able to manipulate content that is common to the various components in virtual space. The devices can communicate through the VirCA platform as well, by sending messages to each other. The logical connection is based on this communication between components. [8]

The operation manager component is connected through the virtual space with the wrapper components for individual cyber devices. Wrapper components are special cyber devices which cannot be manipulated directly by the user. On the one hand, they communicate with the operation manager in a pre-determined semantic form i.e., the operation manager can send requests to wrapper components, which would typically respond by sending back state information as well as information regarding request results (whether or not they were accepted and executed successfully) to the operation manager. On the other hand, they communicate with individual cyber devices using their normal language, i.e. they are typically used to emulate some user interaction. In order to function properly, wrapper components need to know the interaction protocol of each cyber device, as well as the specific events they need to trigger in order to perform some operation.

In the demo scenario, we were using cyber devices for humanoid NAO robots and industrial KUKA robots. The cyber devices had been developed formerly by developers of the 3D Internet based Control and Communication (3DICC) laboratory. We did not alter these cyber devices, but wrapped them up through separate wrapper cyber devices into semantic devices capable of being semantically controlled by the operation manager.

In contrast to wrapper components, the operation manager component is capable of directly communicating with the user. This can happen directly through the default input devices of the VirCA platform, e.g. the voice recognition component. The communication can happen through special input devices as well, which can be developed for special user needs. The special input devices use the standard input interfaces of VirCA. There is no restriction for their operation, as arbitrary input devices can be developed.

3 On the fly solutions

3.1 Matching of devices

If assisted living technologies are applied at the home of elderly persons, one possible solution is to replace the existing devices with intelligent ones, e.g. to replace a normal refrigerator with one that can keep track of what it contains, open its door automatically, etc. One other possible solution is to use the existing devices with other intelligent devices, e.g. mobile robots, which can use these devices designed for human usage, i.e. perform actions and motions like humans. If the latter possibility is chosen, then it also becomes necessary to develop devices that can play the role of the human user.

One other aspect is that applied devices cannot be standardized in the sense that devices may differ from household to household. Because of this, the concrete configuration for the execution of a given operation cannot be determined a priori. There is need for an operation manager service, which looks for the available devices in the intelligent space and gathers a list of their functionalities. If a request for an operation occurs, the operation manager tries to decompose the operation into parts in such a way that the parts can be performed by the available devices in the intelligent space. The operation manager component relies on the description of the capabilities of the devices. Because of the exponential number of possible configurations, it cannot be determined a priori that a single configuration will best suit some given operation. So, if a configuration is found, it has to be executed before it is possible to decide whether the configuration is operable or not.

If a configuration is found to be operable, only then is it confirmed for further use. If a configuration is not operable, the original descriptions of the devices are extended in order to prevent the future decomposition of the same operation into the same configuration.

In the demo scenario, a transportation problem is introduced. There are NAO and KUKA robots, which can transport the medicaments of an elderly woman, and the NAO robots can transport the stick of her. In the first demonstration, the medicaments need to be transported from one room to another, which cannot be solved by a single robot. For the solution of this problem, transportation chains are introduced. Not only one robot can transport an object from a starting to a destination point, but more robots in a chain as well. The first robot of the chain can transport the object from the starting to an intermediate point and so on, the last robot of the chain from an intermediate to the destination point. It can be inferred, to which points one after the other an object can be transported, and which robots one after the other are required for the transportations. This inference over the ontology representation with the help of some inference engine happens in the conceptual space, i.e. this universal intelligence is not bounded to the single devices. If the applied devices are replaced with other ones, the intelligence in the conceptual space remains the same.

3.2 Communication

The availability of individual devices in the intelligent space is also not known a priori. The operation manager should therefore scan for semantic wrapper devices on-the-fly and query the capabilities of the devices connected to them. The cumulative capabilities of the semantic intelligent space are as a result subject to constant change. The configuration of the space can be dynamically changed, devices can be inserted and removed. Because of the architecture of VirCA, communication does not need to be managed by the single components. If some device is added to the virtual space, it can be automatically addressed through VirCA.

VirCA can be considered as a space for knowledge sharing, "market of functions" as well. The single RT-Middleware components are not connected together directly, only through VirCA. Their knowledge is shared in a semantic form, which servers as the basis of the problem solving inference. VirCA helps to realize a collaboration between the devices connected to it in a non-semantic form. With the semantic extention introduced in this paper, this mediation between the needs and the capabilities can be enhanced.

3.3 Matching of ontologies

If semantic information is used for the description and the operation of devices, different devices can use different representations, i.e. different low-level ontologies can be applied for the same target scope. In this case, some kind of mediation is needed between the different ontologies in order to provide for efficient inference, i.e. to be able to resolve user requests.

The high-level VirCA core ontology can be extended with the low-level ontologies of individual devices. The special classes and properties of the low-level ontologies are inherited from the classes and properties of the high-level ontology. Two subclasses of two different ontologies can be identical, disjoint, intersecting or in a class-subclass relationship. The relationship between the two classes should be determined on the basis of the natural language description stored in their ontologies using conventional, non-semantic methods. User supervision is definitely needed in this process. The user can either choose from the proposition of the system or set the relationships manually. The results of this matching process are translator functions between the single devices and the whole ontology.

4 Build-up of the framework

4.1 Components

The framework for assisted living consists of two main parts, the components for the interaction between the user and devices, and the components for ontology management. [Fig.1] The components of the former part are standard components in the VirCA platform, which are connected to the VirCA core component through standard interfaces. The interaction with the devices connected to VirCA happens through the wrapper components which establish a semantic connection between the single devices and the operation manager component. The wrapper components have two main functions: they forward operation requests from the operation manager to the device and send back state updates from the device to the operation manager.

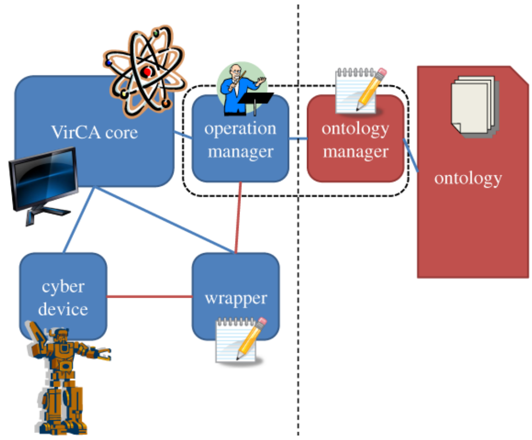

Fig. 1. Architecture of the framework.

The interaction with the user happens logically through the operation manager component, but the operation manager has no own physical interface to the user. Therefore, the user can either use the visual interface of the VirCA core component to pass on user requests through the menu system, or use other arbitrary input components, e.g. speech recognition, which establish a semantic connection between the user and the operation manager. In the reverse direction, the operation manager can use either the built-in functions of VirCA to send information back to the user, or other arbitrary output components, e.g. emotion display, which are in physical connection with the user. The operation manager component has four main functions: on the one hand, it receives user requests from the user through input devices and sends back information about their execution through output devices; on the other hand, it sends operation requests to the devices and receives back state updates from them.

The connection between the platform of intelligent space and cognitive infocommunications (CogInfoCom) is established in such a way that the information flow through the input and output devices is extended with non-verbal information as well. [9] In the conventional approach, some representation of verbal information is communicated between the user and the components, e.g. image of text, sound of speech, including some simple matchings, e.g. clap means turn the light on, ring means cooling water is too hot, etc. In the CogInfoCom approach, input devices are applied to retrieve more information from the user while output devices are applied to pass on more information to the user. [Fig.2] The non-verbal information is not directly related to the verbal information. For instance, in the case of voice recognition, besides the recognition of the command, the sound tone of the user can also be processed and information related to the emotions of the user can be forwarded to the operation manager. In the case of free text voice recognition, based on the method of dictionaries for different styles, the information related to the style of the user can be forwarded as well. [10] Back to the user, output devices can communicate non-verbal information as well based on the state not of the single devices, but the whole system. The operation manager matches the output devices with the dimensions of the system state based on the capabilities of the devices.

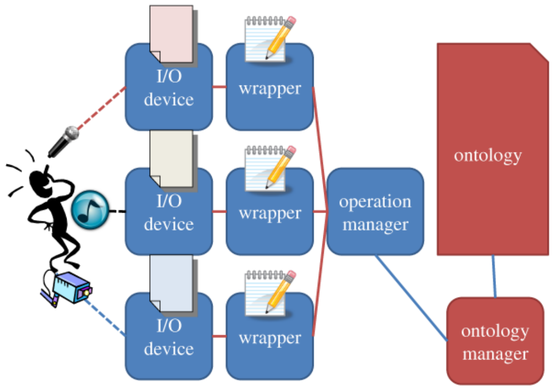

Fig. 2. Cognitive info-communications channels.

The connection between the abilities and state updates of the devices, the state updates of the virtual space, the user requests and the user feedbacks is established in the ontology, the decomposition of the needs into operation of the single devices happens in the ontology based on the state of the whole intelligent space together with the devices. Therefore, an ontology manager component is required as an active element which manages the updates and retrievals of information, inference on the instances and special actions. The operation manager component is connected to the ontology manager component. The operation manager sends state updates, device descriptions, user requests and other user inputs to the ontology manager together with retrieval triggers, i.e. what kind of information is expected as a result. The ontology manager sends back operation requests, user feedback and other user output to the operation manager. The functions of the operation manager and the ontology manager are not logically separate, the two components are two interfaces of one logical unit. It is rather so that the operation manager is more VirCA oriented while the ontology manager is more ontology oriented.

In the demo scenario, the VirCA core itself, the cyber devices for the robots, the wrapper components and the operation manager are standard Windows executable RT-Middleware components, while the ontology manager component is a Java executable RT-Middleware component, as it uses the Protege-OWL Java API to access the ontology. In the state of the development, some facts are inserted manually into the ontology, e.g. the need that the human user - an elderly woman - should take her medicaments and the capability that she can walk only with her stick. The capabilities of the robots are inserted as well concerning the transportation of the objects. Based on the facts in the ontology, an inference is made in order to find possible solutions. The ontology manager scans the ontology for transportation actions, which are needed to be and can be performed as well. If such an action of transportation is found, the ontology manager instructs the operation manager to perform the action. The operation manager instructs the single wrapper components of the robots, which translate the commands to the own "language" of the cyber devices. After the operations had been performed, the wrapper components send back state updates to the operation manager, which forwards them to the ontology manager, which updates the ontology based on them.

4.2 Ontology

For the ontology representation, the OWL-DL of OWL 1.0 was chosen, because it is expressive enough for usage, but still simple enough for inference. For some general framework for editing and inference, the Protege platform was chosen, because its very user and developer friendly operation. [Fig.3] The OWL-DL representation was extended with the use of first order logic in the form of SWRL rules, with the help of which effective inference can be realized over the ontology. For this inference, the Jess rule engine was chosen, which can easily be integrated with the Protege platform. [Fig.4] In the state of the development, the inference was performed in the Protege platform itself, then the result was saved into the ontology, which was scanned by the ontology manager component.

The ontology of the framework for assisted living consists of the general VirCA-core ontology on the one hand and of a human ontology developed for assisted living on the other. The VirCA-core ontology is applied for the matching of operation requests and suitable devices, i.e. the decomposition of high-level user requests into low-level tasks, which can be performed by the devices available in the intelligent space.

Therefore, the VirCA-core ontology contains concepts for processes, which can be required by the user and which can be performed by the devices. [Fig.5] The former is expressed with the concept of need related to some process and agent, and the latter is expressed with the concept of ability related to some process and agent. Based on the a priori information stored in the ontology about the devices which are available at the moment, an on-the-fly decomposition of the user request can be performed. The solutions of this decomposition are process

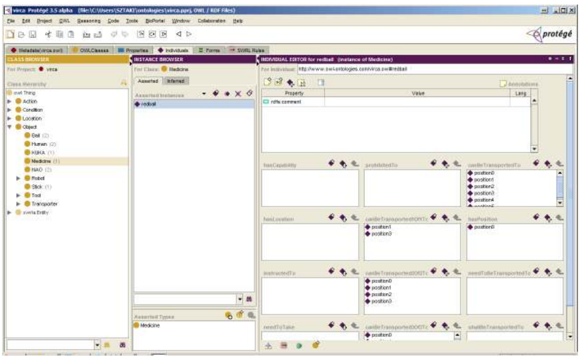

Fig. 3. Inferred properties of the instances in the Protege editor.

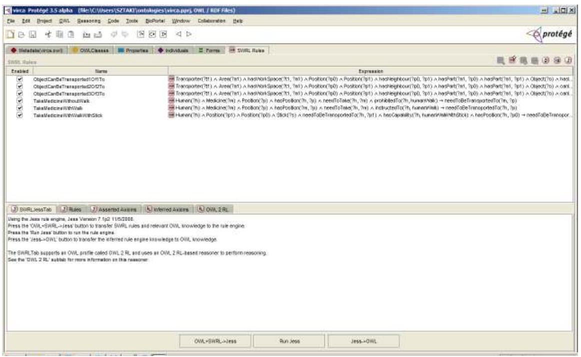

Fig. 4. Running the Jess inference engine in the Protege editor.

configurations. If there are possible solutions, the appropriate configurations will be connected to the concept of the need as possible solutions of the user request.

The human ontology developed for assisted living contains concepts for the description of the mental, emotional and intentional state of the user. For this, the concept of attitude is introduced, which represents the relation of the agent towards some arbitrary object. The attitude can represent some knowledge, assumption, belief of the user towards some expression, which is some proposition about entities of the target scope. The attitude can represent emotions of the user towards entities of the target scope. Emotions can be basic emotions, such as joy, sadness, anger and fear, but can also be higher-level, complex emotions based on lower-level emotions. The attitude can represent intentions of the user towards some change of the virtual space. From these intentions, specific requests of the user can be derived, but not every abstract intention leads to a specific request, and any specific request can be the result of several single intentions.

In the demo scenario, the capabilities were represented that the KUKA and NAO robots could transport the medicaments, and the NAO robots could transport the stick of the elderly woman in their workspace. The need of taking the

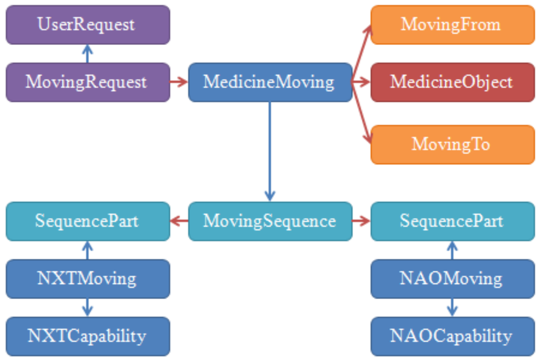

Fig. 5. Semantic decomposition of an action.

medicaments and the walking capabilities of the elderly woman were represented as well together with the priorities based on the instructions of the doctor concerning the amount of walking of the lady. The representation was extended with the possibilities of chain transportation, i.e. whether more robots can transport some object through intermediate places.

5 Application example

For the presentation of the capabilities of the system, a simple demo scenario was compiled. [Fig.6] In the scenario, the location is a flat, where there are stairs between two rooms, which have an offset of height. There are two humanoid NAO robots in the scenario, one in the "upper" room and one in the "lower" room, which can transport light objects in their hands. Unfortunately, the NAO robots are not smart enough to climb the stairs, because it requires too complex algorithms. For that very reason, there are two industrial KUKA robot arms at the stairs, which can grip small objects and transport them between the two floors of different height, i.e. between the two rooms.

In the scenario, the human - who is attended by the intelligent space - is an elderly woman, who is found in the upper room. The situation is, that she should take her medicaments at a formerly defined time. The medicaments are located in the lower room. The NAO robots and the KUKA robots can transport the medicaments as well, so they can be transported to the old woman, if it is required. The capabilities of the elderly woman are represented, e.g. she can walk only with a stick. The sanitary instructions of the family doctor of the elderly woman are represented as well, e.g. whether she should walk a little with a stick or rather not walk at all.

If it is time for the elderly woman to take her medicaments, the system infers that the elderly woman and her medicaments need to be at the same place. If

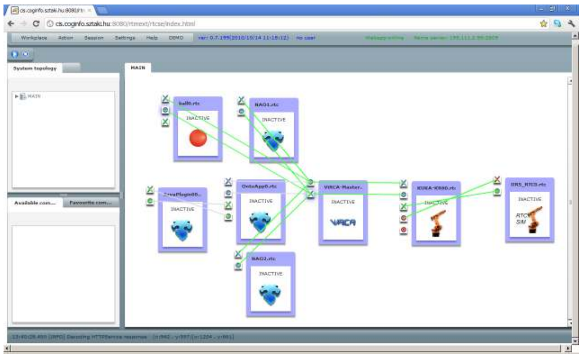

Fig. 6. Demo scenario in the RT-Middleware editor.

the doctor instructed the elderly woman not to walk, the systems infers that the medicaments need to be transported to her. But if the doctor instructed her to walk a little, the system infers that she needs to walk to the medicaments. Based on the fact that the elderly woman can walk only with a stick, in the latter case, the system infers that her stick needs to be transported to her. The stick is found in the upper room, but at a different place.

In the scenario, the facts are inserted that the NAO and the KUKA robots can transport the medicaments of the elderly woman, and the NAO robots can transport the stick of her. The locations of the robots, the medicaments, the stick and the elderly woman are inserted as well. The single robots alone cannot transport the medicaments to the lady as the NAO robots cannot climb the stairs, but the inference can be made in the conceptual space that a transportation chain can be constructed of the single robots, i.e. the NAO robot in the lower room can transport the medicaments to the stairs, one of the KUKA robots can take them from the bottom of the stairs and put them to the top, so finally the NAO robot in the upper room can transport them to the lady.

In the first demonstration, we insert the fact that the doctor forbids the lady to walk too much, and the fact that she should take her medicaments. It is inferred that the medicaments need to be transported to the lady, and a chain of the NAO and KUKA robots can transport them to her. Based on this result, the ontology manager component instructs the operation manager to perform the chain of transportation, so the operation manager instructs the single robots of the chain the perform their single actions. Therefore, the medicaments are finally transported to the lady.

In the second demonstration, we insert the fact that the doctor instructs the lady to walk, and the fact again that she should take her medicaments. It is inferred that the lady should walk to the medicaments, her stick need to be transported to her, and a NAO robot can transport it. Based on this result, the ontology manager component instructs the operation manager to perform this transportation, so the operation manager instructs the NAO robot to perform the action. Therefore, the stick is finally transported to the lady.

6 Conclusion

The intelligent space approach presented in this paper was successful in the sense that the general intelligent space platform could be adopted to the special requirements of assisted living. The use of semantic information demonstrated that in the case of dynamic, on-the-fly solutions, the semantic extension can enhance the current system performance and intelligence. In the field of cognitive infocommunications, further researches would be needed to determine what kind of alternative communication channels could enhance the use efficiency and ergonomy. The possibility is still there to apply the results of these future research in the intelligent space framework.

Acknowledgment

The research was supported by the Hungarian National Development Agency, NAP project, (NAP-1-2005-0021, KCKHA005, OMFB-01137/2008).

References

1. Ambient assisted living joint programme, `http://www.aal-europe.eu/`
2. Domeo project - domestic robot for elderly assistance, `http://www.aal-domeo.org/`
3. Elder spaces project, `http://www.elderspaces.eu/`
4. AALuis - ambient assisted living user interfaces, `http://www.aaluis.eu/`
5. Ando, N., Suehiro, T., Kitagaki, K., Kotoku, T.: Rt (robot technology)-component and its standardization - towards component based networked robot systems development. In SICE-ICASE International Joint Conference, pp. 2633–2638. Busan, Korea, (2006)
6. Galambos, P., Baranyi, P.: Virca as virtual intelligent space for rt-middleware. In IEEE/ASME International Conference on Advanced Intelligent Mechatronic. Budapest, Hungary (2011)
7. Galambos, P., Fülöp, I.M., Baranyi, P.: Virtual collaboration arena: Platform for research, development and education. Acta Technica Jaurinensis, vol. 4, no. 1, 145–155 (2011)
8. Fülöp, I.M.: Extending intelligent collaboration using semantic information in the virca system. In 9th IEEE International Symposium on Applied Machine Intelligence and Informatics, pp. 237–242. Smolenice, Slovakia (2011)
9. Csapó, Á., Baranyi, P.: An application-oriented review of coginfocom: the state-of-the-art and future perspectives. In 10th IEEE International Symposium on Applied Machine Intelligence and Informatics. Herl'any, Slovakia (2012)
10. Fülöp, I.M.: Emotion dependent dialogues in the virca system. In 2nd International Conference on Cognitive Infocommunications. Budapest, Hungary (2011)