Methodologies and scenarios

Application of active index to the management of e-learning activities

Paolo Maresca[a], Shi-Kuo Chang[b] and Michele Pesce[c]

[a]Dip. di Informatica e Sistemistica – Università di Napoli «Federico II» – Naples, Italy; [b]Dep. of Computer Science – University of Pittsburgh, Pittsburgh, U.S.A.; [c]TIM – Telecom Italia Mobile - Naples, Italy

paomares@unina.it; chang@cs.pitt.edu; michele1.pesce@telecomitalia.it

Abstract

An active index consists of a network of index cells. This paper demonstrates the application of active index to the management of e-learning activities in e-learning courses. An e-learning system is a distributed intelligence system where the instructors and students are intelligent human beings. What we want to demonstrate is how to augment human intelligence by an active index. Further research will demonstrate how sensors, mobile communications and search engines can be incorporated into the distributed intelligence system for e-learning.

Edizioni Erickson – Trento
www.erickson.it

Je-LKS Journal of e-Learning and Knowledge Society — *Vol. 2, n. 3, novembre 2006 (pp. 331-341)*

1. Introduction

There are several key recent advances in information technology that will greatly impact our lives. The first one is *mobile communications*, which enable people to communicate anywhere anytime. The second one is *sensor networks*, which allow people to monitor events anywhere anytime. The third one is *search engines*, which allows people to search for information anywhere anytime. These technological advances together or by themselves provide new opportunities in many application areas. New challenges are also created. One central challenge is *how to bridge the gap between communications and intelligence*. With all these recent advances in communications technologies, can we design more intelligent information systems?

The conventional viewpoint is that, in order to design intelligent systems in a distributed computing environment advanced techniques in artificial intelligence, is needed to bridge the gap between communications and intelligence. However we want to make the following observation, which can be confirmed in our daily living in a technology-oriented society: with the advances in sensors and sensor networks, *situated computing* is quickly becoming a reality. For example a car equipped with GPS knows how to behave when it approaches a toll gate. As a consequence of situated computing, hard problems requiring advanced techniques in artificial intelligence can now be substituted by simpler problems, each of which can be solved by distributed intelligence where only a low level of intelligence is provided for each object at each node.

A distributed computing system is therefore also a distributed intelligence system. The key application areas of such distributed intelligence systems include e-learning, tele-medicine, digital library and community network.

In our approach for designing distributed intelligence systems, each object is enhanced by an index cell. Such an object is called a tele-action object. Index cells behave like agents, however there can be numerous index cells. Objects enhanced by index cells can perform actions by themselves. Therefore, intelligence is distributed to these tele-action objects. Objects may also contain multimedia data.

Active index consisting of index cells can best be used in a distributed intelligence system. A lot of distributed knowledge can increase the overall intelligence of an intelligent system. As an example, the infrastructure of a crime-prevention community network can be based upon a distributed computing system equipped with sensors, microbots, GPS and active index. The crime-prevention community network supports the detection of significant events, by creating and maintaining relationships among multimedia objects.

In this paper, we will demonstrate the application of active index to the management of e-learning activities in e-learning courses. An e-learning system is also a distributed intelligence system, where the instructors and students are intelligent human beings. What we want to demonstrate is how to augment human intelligence by an active index. The fundamentals of the index cell will be explained in section 2. In section 3 the use of index cells in e-learning. Section 4 offers some concluding remarks.

2. The Distributed Intelligence System's Basic Element: The Index Cell

The following is a short *three-way Index Cell (IC) description* (Chang, 1995; 1996; 2006) in order to explain what IC is and what it can do.

Three different points of view of the same subject: the most important is, of course, the mathematical one that is the formal definitions embracing all the possible descriptions. The others two are given to show how wide could be the applications domain related to the use of the Index Cells.

2.1 The Index Cell as a computational model

The Index Cell is a particular Finite State Machine which accepts Input messages, executes operations and sends one or more output messages to one or more IC or to external environment.

The amount and type (or types) of IC depends on the state and Input Messages. It is a Mealy model machine and, according to the problem domain, could be deterministic or non-deterministic, but as theory states, any ND-FSM could be transformed in a deterministic FSM.

2.2 The Index Cell as a Knowledge detector

The Index Cell is an *intelligent agent*, it perceives the environment through the messages and executes the actions in order to reach the goals. The goals represents the knowledge acquiring in the environment where the cell acts.

2.3 The Index Cell as a mathematical model

An *index cell* is described by $ic = (X,Y,S,s_o,A,t_{max},f,g)$ where:

- X is the (possibly infinite) set of input messages including dummy input *d;*
- Y is the (possibly infinite) set of output messages including dummy output *d;*
- S is the (possibly infinite) set of states. S includes a set of ordinary states **S** and a special state s_{dead} called the dead state. If an index cell is in the dead state, it is a dead index cell, Otherwise it is a live index cell. s_o in **S** is the initial state of the index cell *ic;*
- A is the set of action sequences that can be performed by this index cell*;*
- t_{max} is the maximum time for the cell to remain alive, without receiving any messages. If t_{max} is infinite, the cell is perennial*;*

- f is a function: 2^X x S -> {0,1} where 2^X is the power set of input X,
- g is a function: 2^X x S -> 2^{ICB} x Y x S x A.

The following figure shows an example of a generic index cell named IC deficiency cell inside a generic IC network composed of other two Index cells IC Proficiency Cell and IC self adjustment cell. It is a particular module 2 counter with a reset functionality.

This Index Cell is analogous to the proficiency cell, the only difference resides in the number of states necessary to realize the counter (3 in the proficiency cell). In this case two single states d_0, the initial state, and d_1 are necessary. Particularly d_1, indicates the access to the document of deficiency from part of a single student. In the state transition $d_0 \rightarrow d_1$, as it happens for the Proficiency Cell, the deficiency cell sends a message of output to the Professor Cell, in order to notify the regression of the virtual class and a message of reset to the Proficiency Cell, to the aim to reset the counter. Differently to Proficiency Cell, this cell sends to the Self-Adjustment Cell the message *easier*, in such way to impose the distribution of one lesson with a smaller level of difficulty.

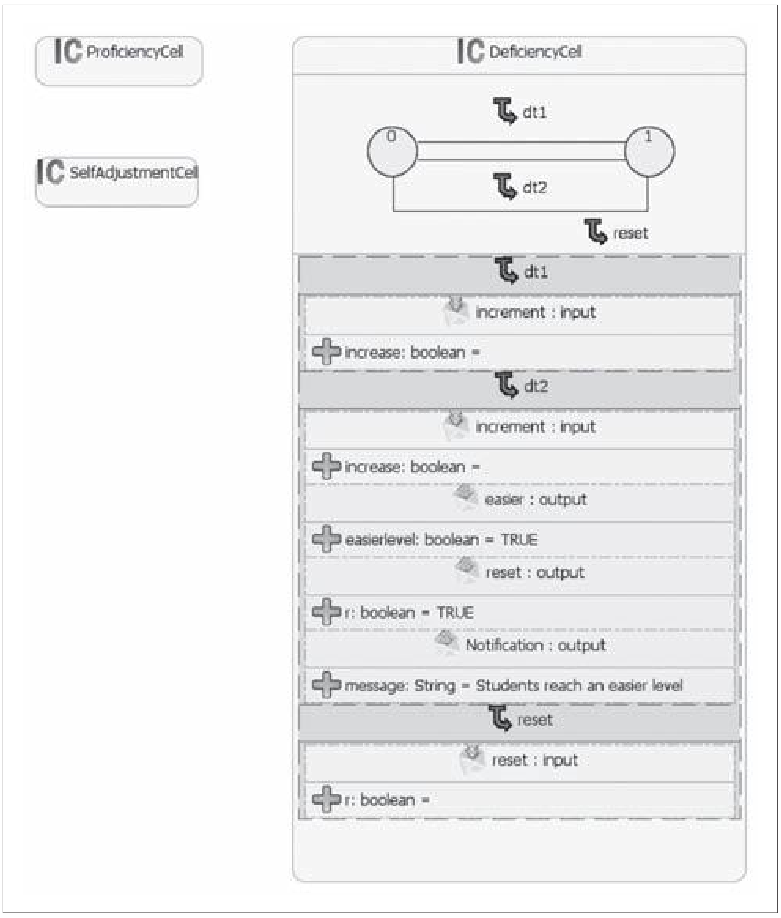

Figure 1 Deficiency Index cell detail inside a generic ICs network.

3. Index cells and e-learning

Index cells could be used in order to manage e-learning activities in e-learning courses (Arndt, 2002; 2003; 2007; Maresca, 2003).

An e-learning system is a distributed intelligence system, where the instructors and students are intelligent human beings. Some kind of *intelligent behavior* can be identified and synthesized into index cells. Particularly managing e-learning activities some *intelligent behavior* could be the followings: (i) proficiency, (ii) deficiency. For the sake of the brevity we will discuss only the first two *intelligent behavior*. The proficiency and deficiency definitions will be given in the next paragraph while discussing an example.

Index cells are also useful to express non functional requirements.

3.1 The proficiency-deficiency behavior management example

This chapter is aimed to give a design-oriented solution to the exercise 2 of the CS2310 course (Chang, 2006a; 2006b). The problem can be synthetized as follows.

Let us consider an e-learning adaptive system. Lessons are organized through one hypermedia structure. Such structure remains passive until it does not come visited from the students. Such structure can be rendered active associating the index cells with multimedia documents. The idea is to construct a special document that, if visited from some students, give us notification that they have caught up a sure level of proficiency and therefore the learning materials would have to be modified in order to become more difficult. In the same way, when some students approach to a special document that indicates deficiency, it means that they have found problems and consequently the learning materials would have to be rendered simpler.

The following types of Index Cells have been specified in order to solve the problem:

Proficiency-level index cell (PLIC): this index cell is associated with a specific multimedia document (reachable only from the students well prepared). When it comes primed, the level of proficiency comes increasing of 1. When the level of proficiency has caught up a predefined threshold (as an example 3), the Index Cell will send to the professor a message, that will inform him that a sufficient number of students has caught up this level of proficiency. The *PLIC* will send also messages to some multimedia documents, to the aim of increasing the level of difficulty.

Deficiency-level index cell (DLIC): this index cell (see also figure 1) is associated with a specific multimedia document (reachable only from the students with deficiencies). When it comes primed, the level of deficiency comes increasing of 1. When the level of deficiency has caught up a predefined threshold (as an example 2), the Index Cell will send to the professor a message, than will inform him that a sufficient number of students has caught up this level of deficiency. The *DLIC* will send also messages to some multimedia documents, to the aim to reduce the level of difficulty.

Self-adjustment index cell (SAIC): this Index Cell is associated to multimedia documents containing e-learning material. When it receives the message *harder*, it increases the difficulty of the learning material. In the same way, when *SAIC* receives the message *easier*, it reduces the difficulty of the learning material to the aim to render them simpler.

The istances of Index Cells are associated to multimedia documents.

The example is composed of one multimedia lesson (figure 4) containing 5 various levels of difficulty. The system distributes a more difficult lesson when three students approach to a multimedia document named *Proficiency Document*. The system distributes a simpler lesson when two students approach a special document named *Deficiency Document*.

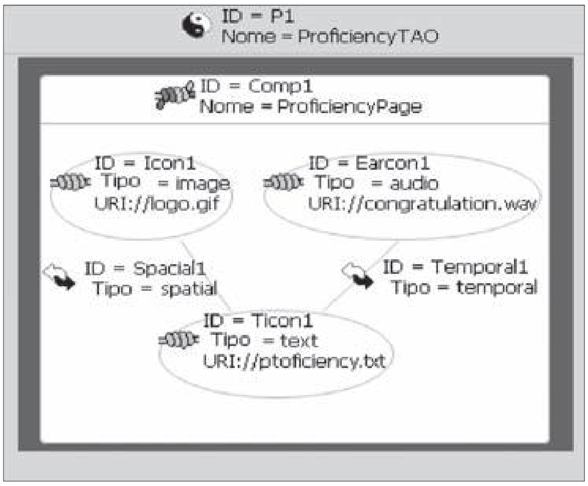

Figure 2 Proficiency TAO.

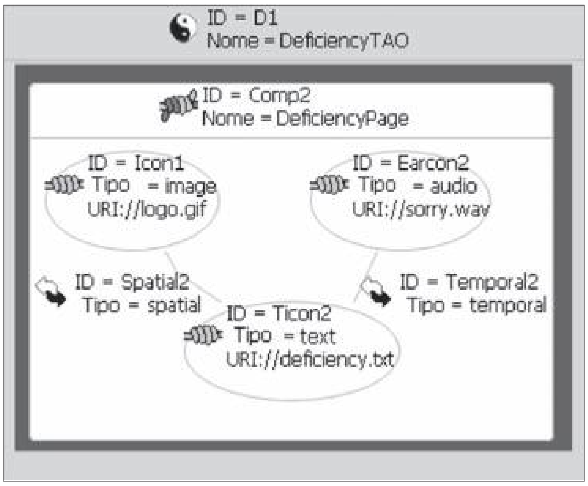

Figure 3 Deficiency TAO.

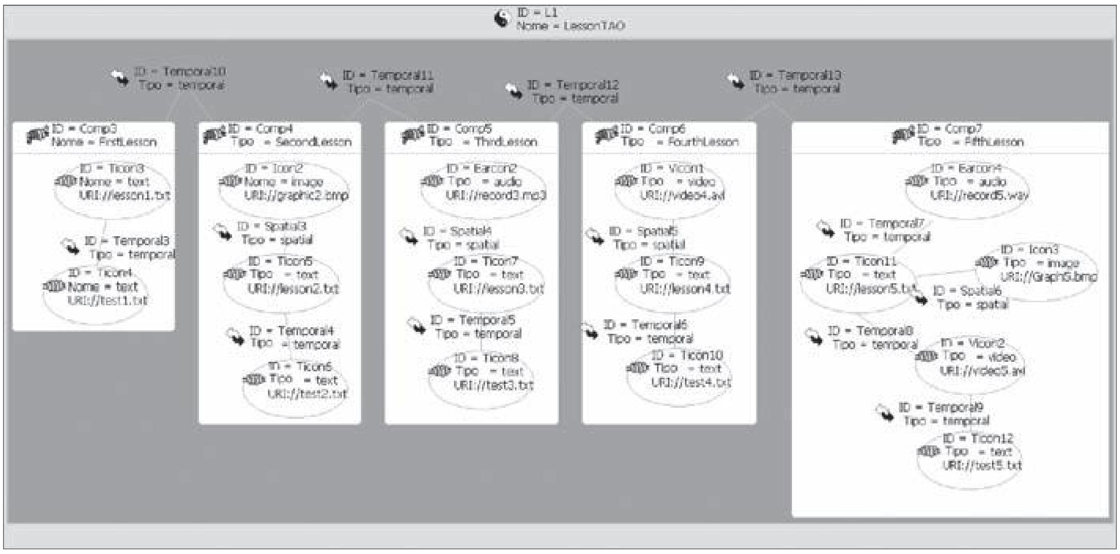

Figure 4 TAO Multimedia Lesson.

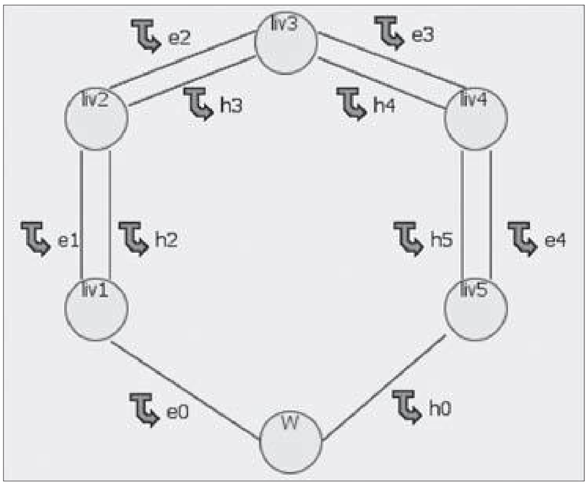

Figure 5 Self-Adjustment Index Cell.

In these hypotheses, the hypermedia structure of the e-learning application is composed of:

(i) *Proficiency Document*: it is a special document (see figure 2), which can approach only the students who have exceeded the annexed verifications to the multimedia lessons with profit. The document communicates to the student its result.

(ii) *Deficiency Document*: it is a special document (see figure 3), which can approach only the students who have not exceeded the annexed verifications to the lessons with a sufficient result. The document communicates to the student its result.

(iii) *Multimedia Lesson*: it is complex a multimedia document (see figure 4) , containing the e-learning material. It is divided into five lessons on the same argument with increasing difficulty.

Only a lesson at once must be accessible depending on the proficiency of the virtual class.

All these three documents are expressed in terms of TAO (Chang, 1995b; Guercio, 1998; 2002).

As example the TAO hyper-graph of figure 2 is composed of a macronode, to the whose inside is defined a *Ticon*, containing a text of congratulations that is shown to the customer, one *Icon*, containing logo of the department and one *Earcon* content an audio message of congratulations. The *Earcon* is tied to the *Ticon* with a temporal link, in order to synchronize the audio execution with the reading of the text. The *Icon* is connected to the *Ticon* using a spatial link that defines the mutual position of the two generalized icons.

The described objects, represent the hypermedia structure of e-learning application. They do not constitute a multimedia system, because it lacks the mechanisms that regulate dynamics of the application. In other words the hypermedia structure need to react to the user input so it is necessary that the static structure of the system will be associated to a knowledge structure. In figure 6 the knowledge network associated to the described multimedia object is showed.

The IC Level1-Level5 represent the knowledge structure associated to multimedia documents representing the lessons. Their task are to send an output increment message, with the parameter increase: boolean=TRUE, to the IC Proficiency, if the student has obtained a positive result from the test. The IC Proficiency will send a message to the IC Deficiency (Boolean=FALSE), if the Student has obtained insufficient score. The IC Professor, receives the notification messages sent from both the IC Proficiency (ICP) and IC Deficiency (ICD). The ICP and ICD have the tasks to notify the Teacher that the level of difficulty of the distributed multimedia lesson is, respectively, increased or diminished; and it receives messages of *Easier Lesson* or *Harder Lesson* from the Self-Adjustment IC (figure 5), when the contents distributed are, respectively, too much difficult or too much simple. Both static structure and knowledge structure are mapped together into an xml (Young 2000) file produced by a Multimedia Knowledge Environment for Index Cell (*MKE*^{2} *4IC*) (Scarfogliero and Sorrentino, 2006).

4. Conclusion

Working on projects, it is easy to understand how fluent can be the work using IC. It is like applying mathematical induction: step by step, starting from the particular problem you can obtain general rules, so that, quite immediately you can discover other specifications omitted or implicit in problem domain and then add

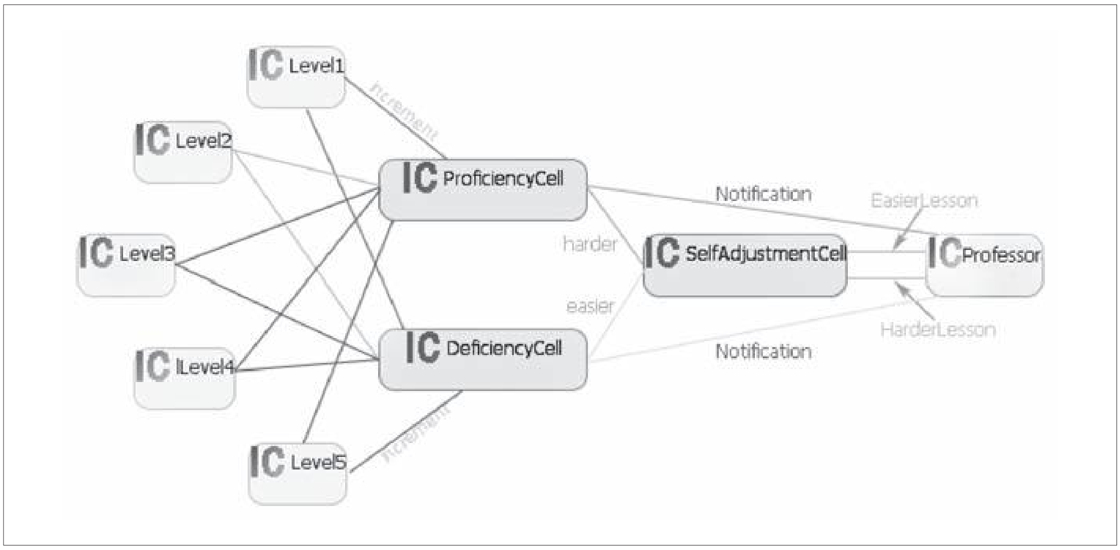

Figure 6 Knowledge structure associated to multimedia objects.

further solution. IC Systems have many powerful features: flexibility, scalability, portability. All of that are key factor

Working on projects, it becomes clear that IC Systems have many powerful features: flexibility, scalability, portability that are essential in Software Engineering and then in Multimedia Software Engineering.

In conclusion we have demonstrated in this paper how to augment human intelligence by an active index. Further research will demonstrate how sensors, mobile communications and search engines can be incorporated into the distributed intelligence system for e-learning.

Acknowledgements

We would like to express our thanks to Scarfogliero Giuseppe Marco and Sorrentino Lorenzo for their effort in the designing and implementing a Multimedia Knowledge Environment for Index Cell (*MKE*^{2} *4IC*) , which is an eclipse environment for modelling a TAOXML private knowledge.

BIBLIOGRAPHY

Arndt, T., Chang S.K., Guercio, A., Maresca P. (2007). *An XML-Based Approach to Multimedia Software Engineering for Distance Learning*. In Future directions in Distance Learning and Communication Technologies, Information Science Publishing, USA.

Booch, G., Rumbaugh J., Jacobson, J., (1999). *Unified Modeling language*, Addison Wesley.

Maciaszek, G., (2001). *Requirement analysis and system design: developing information systems with UML*, Addison Wesley.

Burke, E.M., (2001). *Java and XSLT*, O'Reilly.

Young, J.Y., (2000). *XML step by step*, Microsoft.

Arndt, T., Chang S.K., Guercio, A., Maresca P. (2003). An XML-Based Approach to Multimedia Software Engineering for Disctance Learning, *Journal of Distance Education Technologies* 1 (1): 40-62.

Maresca P., Arndt, T., Guercio, A., (2003). Unifying Distance Learning resources: The metadata Approach, *Journal of Computers* 13 (2): 60-76.

Chang H.J., Hou, T.Y., Hsu A., Chang S.K., (1995b). The Management and Application of Tele-action Objects, *Journal of ACM Multimedia Systems* 3 (5-6): 204-216.

Chang S.K., (1996). Extending Visual Languages for Multimedia, *IEEE Multimedia* 3 (3): 18-26.

Chang S.K., (1995). Towards a Theory of active index, *Journal of Visual Languages and Computing* 6 (1): 101-118.

Davis, M., Denning, K. (2000). Online learning: frontiers in the creation of learning communities. *Proceedings of the Conference: Networked Learning 2000: Innovative Approaches to Lifelong Learning and Higher Education Through the Internet*, University of Sheffield.

Albanese, M., Maresca, P., Picariello A., Rinaldi A.M. (2005). Towards a Multimedia System: an Approach using TAO_XML. *Proceedings of the Conference: Distributed Multimedia Systems 2005*, Banff, Canada.

Guercio, A., Arndt, T., Chang S.K., (2002). A Visual Editor for Multimedia Application Development. *Proceedings of the 22nd IEEE Conference: Distributed Computed System Workshop 2002.*

Arndt, T., Chang S.K., Guercio, A., Maresca P. (2002). An XML-based approach to multimedia software engineering for distance learning. *Proceedings of the 14nd International Conference on software engineering and knowledge engineering, 2002.*

Guercio, A., Arndt, T., Chang S.K., (1998). Visual Tool for a Multimedia IC Development Environment (MICE). *Proceedings of the IEEE Symposium on Visual Language, 1998.*

Chang S.K., *CS2310 Multimedia software engineering*, (documento WWW) URL: <http://www.cs.pitt.edu/~chang/231/231syl.html> accessed on 15th October 2006.

Chang S.K., *IC Builder user manual*, (documento WWW) URL: <http://www.cs.pitt.edu/~jung/IC_Builder/icb_manual.html> accessed on 15th October 2006.

Chang S.K., *IC Builder executable*, (documento WWW) URL: <http://www.cs.pitt.edu/~jung/IC_Builder/icb_tapp.zip> accessed on 15th October 2006.

Chang S.K., *CS2310 Exercise 2*, (documento WWW) URL: <http://www.cs.pitt.edu/~chang/231/231ex.html> accessed on 15th October 2006.

Chang S.K., *Towards a Theory of Active Index*, (documento WWW) URL: <http://www.cs.pitt.edu/~chang/365/ic.html> accessed on 15th October 2006.

Scarfogliero G.M. (2006). *A Multimedia Knowledge Eclipse Environment per le Celle Indice: Progetto ed Implementazione*. Ph.D. Thesis, University of Napoli *Federico II*.

Sorrentino L. (2006). *A Multimedia Knowledge Eclipse Environment per la gestione del TAOXML: Progettazione ed Implementazione*. Ph.D. Thesis, University of Napoli *Federico II*.

Pesce M. (2005). *La conoscenza applicata al TAOXML environment*. Ph.D. Thesis, University of Napoli *Federico II*.

Piscitelli M. (2005). *Un ambiente di ontologie multimediali basato su TAOXML*. Ph.D. Thesis, University of Napoli *Federico II*.

Marmo C. (1999). *Un linguaggio XML.based per i TeleAction Object: Progettazione e Realizzazione di un ambiente TAOXML*. Ph.D. Thesis, University of Salerno.