



FREIE UNIVERSITÄT BOZEN  
LIBERA UNIVERSITÀ DI BOLZANO  
FREE UNIVERSITY OF BOZEN - BOLZANO



Faculty of Computer Science, Free University of Bozen-Bolzano  
Piazza Domenicani 3, 39100 Bolzano, Italy  
Tel: +39 04710 16000, fax: +39 04710 16009, <http://www.inf.unibz.it/krdb/>

*KRDB Research Centre Technical Report:*

Global Temporal Reasoning on Stories with  
LODE:  
a Logic-based E-tool for Deaf Childrenen

R. Gennari, O. Mich

<b>Affiliations</b>	KRDB, Faculty of Computer Science, FUB Piazza Domenicani 3, 39100 Bolzano, Italy
<b>Corresponding author</b>	R. Gennari gennari@inf.unibz.it
<b>Keywords</b>	assistive technology, constraint programming, e-learning tools, evaluation, human computer interaction
<b>Number</b>	KRDB08-4
<b>Date</b>	21-06-2008
<b>URL</b>	<a href="http://www.inf.unibz.it/krdb/">http://www.inf.unibz.it/krdb/</a>

**©KRDB Research Centre.** This work may not be copied or reproduced in whole or part for any commercial purpose. Permission to copy in whole or part without payment of fee is granted for non-profit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of the KRDB Research Centre, Free University of Bozen-Bolzano, Italy; an acknowledgement of the authors and individual contributors to the work; all applicable portions of this copyright notice. Copying, reproducing, or republishing for any other purpose shall require a licence with payment of fee to the KRDB Research Centre.

## **Abstract**

LODE is a web tool for deaf children, which aims at stimulating global reasoning on written e-stories. This paper reports on the design of LODE, and an initial prototype application of LODE. First, we motivate the need of an e-tool such as LODE for deaf children, explain its reasoning exercises and the expected learning outcomes. Then the paper describes the design of the client-server architecture of LODE: the server employs a constraint-based automated reasoner; the client is a web user interface. Finally, we report on the assessment of the initial prototype of LODE with expert users and two deaf children, laying the groundwork for the development and a long-term evaluation plan of LODE.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Preliminaries</b>	<b>3</b>
2.1	Literacy Issues and the Educational Goal of LODE . . . . .	3
2.2	User Interface Design . . . . .	4
2.3	Test Design . . . . .	4
<b>3</b>	<b>Background on Automated Temporal Reasoning with Constraint Programming</b>	<b>6</b>
3.1	Time Representation à la Allen . . . . .	6
3.2	Constraint Programming for Automated Temporal Reasoning . . . . .	7
3.2.1	Constraint modelling . . . . .	8
3.2.2	Constraint solving . . . . .	8
<b>4</b>	<b>LODE: Educational Exercises</b>	<b>9</b>
4.1	Dictionary . . . . .	9
4.2	Global Reasoning Exercises . . . . .	9
4.2.1	Comprehension . . . . .	10
4.2.2	Production . . . . .	10
4.2.3	Final Remarks . . . . .	10
<b>5</b>	<b>The Architecture of LODE</b>	<b>12</b>
<b>6</b>	<b>The Constraint-based Automated Reasoner</b>	<b>14</b>
6.1	Constraint Modelling in LODE. . . . .	14

6.2	Constraint Solving in LODE . . . . .	15
6.3	Why Constraint Programming in LODE . . . . .	16
6.3.1	Constraint modelling . . . . .	16
6.3.2	Constraint solving . . . . .	16
<b>7</b>	<b>The GUI of LODE</b>	<b>18</b>
<b>8</b>	<b>Related Work</b>	<b>23</b>
8.1	Italian Tools . . . . .	23
8.2	Non-Italian Tools . . . . .	24
<b>9</b>	<b>Assessment</b>	<b>28</b>
<b>10</b>	<b>Conclusions</b>	<b>30</b>



# Chapter 1

## Introduction

Good literacy is essential for everyone; it ensures a continuous process of personal maturation and a positive social integration. Deaf children encounter several difficulties in learning to read and write with proficiency; they tend to reason on isolated concepts and not relate distant concepts in written texts. Information technologies can be usefully employed to create tools for their literacy.

In recent years, much research in deaf studies and computer science has been devoted to applications for sign languages; roughly speaking, a *sign language* is a gestural-visual language with signs as lexical units, whereas a *verbal language* is an oral-auditive language with words as lexical units. Less attention seems to be paid to e-learning tools for improving the literacy of deaf children in verbal languages. Our LOgic-based e-tool for DEaf children (LODE) belongs in this latter class. It aims at stimulating children to *globally* reason on a story by relating distant episodes — a step which is necessary for fully comprehending the story and its morale. This is done through apt exercises, developed with the assistance of experts in deaf studies. In its current form, LODE stimulates deaf children to reason on the temporal dimension of children's stories, written in verbal Italian.

The technological core of LODE is composed of a constraint-based automated reasoner; the reasoner can be employed both for creating some of the LODE exercises automatically, and for generating individual feedback from the exercise solutions proposed by the LODE user.

Automated reasoning has been used in intelligent tutoring systems before. See WHY2-ATLAS, an intelligent tutoring system for qualitative physics [1], for example. But the LODE breakthrough is to apply automated reasoning techniques, in particular constraint-based reasoning, to an e-learning tool for children, in particular deaf children, which opens up new possibilities as well as challenges.

This paper describes the general design of LODE and reports on an initial prototype of LODE, tested with expert users and two deaf children; a demonstrator of the tool is available online [2]. Section 2 offers the necessary preliminaries. We start by motivating the need of an e-learning tool such as LODE, that is, we give grounds for its educational goal. Designing and testing tools for and with deaf users present several challenges, and we overview them as we encountered them in the development of LODE. Since the design of LODE requires temporal reasoning with constraint programming, Section 3 provides a gentle introduction to it.

Then the paper delves into LODE, presenting its general design. Section 4 outlines the educational exercises of LODE and how they aim at stimulating global reasoning with temporal relations. The general architecture of LODE is discussed in Section 5; Section 6 is rather technical, it explains in details the role of constraint programming in developing and solving the educational exercises of LODE; the user interface of LODE is instead described in Section 7. We relate LODE to other e-tools for deaf children in Section 8, highlighting its novelty in the landscape of e-learning tools for deaf children.

The initial prototype of LODE was tested and assessed with expert users and two deaf children; Section 9 reports on the results of the evaluation, substantiating the need of an e-tool such as LODE, and proposes a long-term evaluation plan. Section 10 provides some conclusions and suggestions for future work.



## Chapter 2

# Preliminaries

### 2.1 Literacy Issues and the Educational Goal of LODE

Due to a limited exposition to the language in its spoken form in their first years of life, deaf people lack the primary, natural means of acquiring literacy skills: “deaf children have unique communication needs: unable to hear the continuous, repeated flow of language interchange around them, they are not automatically exposed to the enormous amounts of language stimulation experienced by hearing children” [3].

Research findings point out that deaf people hardly achieve verbal language literacy; they tend to reason on single episodes and show difficulties in formulating coherent global relations, such as temporal relations, between episodes of narratives in a verbal language [4], to the effect that their ability of reading does not often go beyond that of an eight-year old child [5]. As reported in [4], this attitude can also depend on the kind of “literacy interventions addressed to deaf children” which tend to “focus on single sentences and the grammatical aspects of text production”. A novel literacy e-tool for them should thus *stimulate global deductive reasoning* on entire narratives, written in a verbal language such as Italian. This is the main educational goal of our e-tool.

To this end, LODE presents children with e-stories and invites children to globally reason on each of them; that is, children are elicited to analyse and produce relations between events, even distant in the story, so that the relations are consistent with the story and possibly implicit in it.

LODE focuses on a specific type of relations, namely, temporal relations. Temporal dimension is a concept that children learn indirectly through narration [6]; this may explain why it tends to be a difficult concept for deaf children, who are not exposed to oral narration as hearing children are [4]. Every text can be imagined as a representation of a chronological order of their world’s events. The author decides the order to recount the events of a story. This order can be *natural* or *artificial*. It is natural when the textual order is equal to the real order; in this case, events are organised only with the operator *and then*. It is artificial when text redistributes the real events: it can, for example, put at the beginning of the text an event that in reality happens as the final one. In the case of artificial order, the operators used are *and before* or *and in the*

*meantime*. Children learn the *and then* operator at around the age of four. They learn the *and before* operator at around the age of five to seven. They learn the *and in the meantime* operator last. Although children easily learn the *and then* operator, they need more time for mastering the other operators.

Telling stories to children is a way to help them learn operators that organise time, and enhance children's ability of comprehending written texts and writing narratives [7]. This is why we decided to design LODE around famous children's stories, stressing the temporal aspects of the stories; LODE narrates temporally rich stories for children and then stimulates children to create a coherent network of temporal relations out of each story.

## 2.2 User Interface Design

The unique characteristics of deaf users and the high variability inside this group of users requires sensitivity and special attention in designing user interfaces for them. Specific usability and accessibility rules should be followed. Clearly, in the design of the user interface of e-tools to be used by deaf people, such as LODE, the visual input should always augment or replace the auditory input [8]. Moreover, captions must be provided with all multimedia presentations and all visual cues must be noticeable even if the user is not looking straight at the screen [9].

According to some research findings [10], deaf and hearing people encode information differently, with different strategies in the organisation and access to knowledge stored in the work and long term memory; deaf people seem to focus mostly on concept details and images rather than on relations among concepts when processing information. The use of graphics and hypertext in e-tools for deaf people should consider such findings [11].

Last but not least, LODE is for children, thus graphics plays a relevant role in it. However, non-standard graphical interaction techniques could cause predictable problems in hypertextual user interfaces; lack of perceived clickability affordances, such as overly flat graphics, cause users to miss features because they overlook links [12]. As Nielsen says [13] "the idea that children are masters of technology and can defeat any computer-related difficulty is a myth. [...] Poor usability, combined with kids' lack of patience in the face of complexity, result in many simply leaving web sites".

## 2.3 Test Design

Testing software applications *for* and *with* deaf children requires particular care, and specific methods.

When evaluating e-tools with deaf users, several methods for usability testing [14] cannot be directly employed. For instance, due to the aforementioned literacy problems, traditional inquiry methods based on questionnaires or oral interviews to deaf users are rather impracticable. The evaluation could be simplified testing the e-tools not with deaf users, but with hearing adult experts [15]. Another procedure is that of involving in the experiment not only deaf children but also their parents or their therapists (e.g., sign language teachers) working as interpreters [16]. In this case, the influence of the presence of the interpreters on the procedure must be observed in details [17]; in fact, their mediation can easily lead to imprecise results.

In general, it is important to describe what it is like to participate in a usability test to the session participants before starting the test [14]. In the case of deaf users, it is not always possible for test designers to communicate directly to participants, and interpreters may misunderstand the explanation of the test procedure.

Another method consists in recording users on videos and their actions on a log file; the video could show the facial expressions and the tester can then perceive, for instance, which of the logged tasks are critical for deaf children. However, interpretation errors can be easily introduced in this manner.

Last but not least, recruiting deaf users willing to test e-tools for them is another main problem; deaf people are usually spread on the territory, and, due to a long history of isolation, they generally tend to be distrustful of hearing people [18].

## Chapter 3

# Background on Automated Temporal Reasoning with Constraint Programming

In order to assist the child in deducing consistent relations between temporal events of a story, LODE will employ an automated temporal reasoner. Automated Temporal Reasoning is a branch of Artificial Intelligence. An instance of a temporal reasoning problem is given by the following exercise of LODE; the excerpt is taken from a simplified version of *The Ugly Duckling* by H.C. Andersen.

Mummy duck is sitting on some eggs: she has five eggs, four are small, and one is big. All of a sudden, while she is still sitting on eggs, the small eggshells crack and four little yellow ducklings peep out. Mummy duck watches the big egg but sees no signs of cracking. . . So she decides to keep on sitting on it. After some days, while she is sitting on it, the big eggshell also cracks and an ugly gray duckling peeps out. . .

**Exercise:** *do the small eggshells crack before the big eggshell cracks?*

Answering such a question means solving a temporal reasoning problem; solving it in an automated manner means choosing a formal representation of time and a computational automated reasoning system for this. We use the problem in the remainder to explain the time representation, in Section 3.1, and the computational reasoning system, in Section 3.2, that LODE adopts.

### 3.1 Time Representation à la Allen

In the Allen time representation, intervals are the primitive entities for representing time; each interval is uniquely associated with a time event. Between any two pairs of events, there is an *atomic Allen* relation, namely, a relation of the form

before, meets, overlaps, starts, during, finishes, equals

or  $\text{rel}^{-1}$ , where  $\text{rel}$  is one of the above relations and  $\text{rel}^{-1}$  is the inverse of  $\text{rel}$ . See Fig. 3.1 for an intuitive graphical representation of the atomic Allen relations between two events,  $e1$  and  $e2$ . The Allen relations are employed whenever temporal information boils down to qualitative

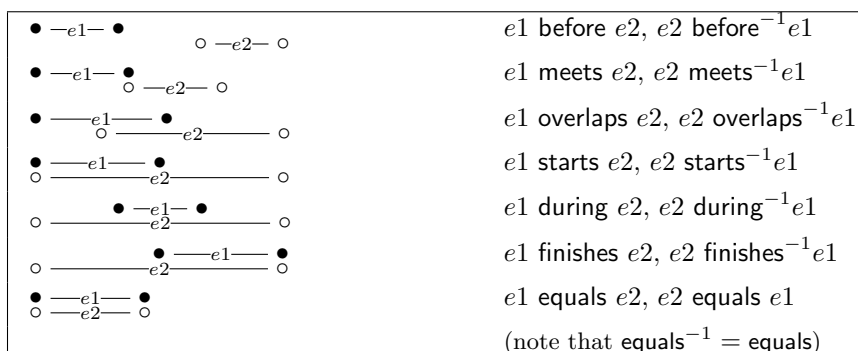


Figure 3.1: The atomic Allen relations.

relations between events, e.g.,

$$\text{“the small eggshells crack while Mammy duck sits on some eggs”}; \quad (3.1)$$

in terms of the Allen relations, the sentence (3.1) states that the relation *during* can hold between the event “the small eggshells crack” and the event “Mammy duck sits on some eggs”. By way of contrast, “the small eggshells crack *after 2 days*” provides a quantitative temporal information.

As Allen arguments [19], “uncertain information” can be represented by means of unions of the atomic Allen relations. Formally, let  $A$  denote the class of the atomic Allen relations. Then the class  $2^A$  forms a relational algebra with the operations of union  $\cup$ , composition  $\odot$ , and inverse  $^{-1}$ . In particular, the composition operation is used in deductions with the Allen relations. For instance, the result of the composition *before*  $\odot$  *before* is *before*; this means that

$$\begin{array}{ll} \text{if} & e1 \text{ before } e2 \text{ and } e2 \text{ before } e3 \\ \text{then} & e1 \text{ before } e3. \end{array}$$

## 3.2 Constraint Programming for Automated Temporal Reasoning

Constraint programming is at the intersection of logic programming, optimisation and artificial intelligence. The central notion is that of constraint, informally, a relation over variables which restricts the values the variables can take in their domains; for instance, a linear equation over the domain of real numbers is a constraint. Constraint programming is “an alternative approach to programming in which the programming process is limited to a generation of requirements (constraints) and a solution of these requirements by means of general or domain specific methods” [20]. Nowadays it is implemented in several heterogeneous environments, e.g., B-Prolog [21], a proprietary Prolog-based Constraint Logic Programming (CLP) system, ECL<sup>i</sup>PS<sup>e</sup> [20, 22], an open-source CLP system, or GNU Prolog [23], a free Prolog compiler with constraint solving over finite domains.

Even if constraint programming is successfully used in several applications ranging from scheduling to numerical analysis (see for example [24]), to the best of our knowledge, LODE is the first attempt of applying it to develop an e-learning tool for children.

Performing automated temporal reasoning with constraint programming means first *modelling* a temporal problem in a specific formalism and then *solving* it in an automated manner with a constraint programming system. In the following, we introduce the basics on constraint modelling and solving, drawing them from [25].

### 3.2.1 Constraint modelling

. Given the temporal reasoning problem above, the constraint programmer *models* it as a *constraint satisfaction problem*. In essence, this is given by

- finitely many variables,  $x_1, \dots, x_n$ ,
- each ranging on a domain  $D_i$  of values,
- and a set of constraints, namely, relations of the form  $C \subseteq D_{i_1} \times \dots \times D_{i_m}$ .

A tuple of domain values  $a_1, \dots, a_n$  for the variables  $x_1, \dots, x_n$  is a *solution* to the problem if it belongs to all the constraints of the problem. When a value  $a_i$  for  $x_i$  participates in a solution to the problem, we will say that  $a_i$  is *consistent* with the problem.

A temporal reasoning problem can be modelled as a constraint problem as done in [26]. We expound this modelling in details in the context of LODE in Section 5. As an example, take the temporal reasoning problem in p. 6. The ordered pair of events “the small eggshells crack” and “the big eggshell cracks” gives a variable; its domain is the set of the atomic Allen relations; the temporal reasoning problem restricts the domain of the variable to **before**, and this is a unary constraint. The relation **before** is consistent with the considered problem.

### 3.2.2 Constraint solving

. Once a temporal reasoning problem is modelled as a constraint problem in a suitable programming language (e.g., CLP), a constraint programming system (e.g., ECL<sup>i</sup>PS<sup>e</sup>) can be invoked to *solve* it. In this setting, to solve a problem means

- to decide on the consistency of an Allen relation with the problem, e.g., **before** between “the big eggshells crack” and “the small eggshell cracks”,
- or to deduce an/all the Allen relation/relations between two events, e.g., “the big eggshell cracks” and “the small eggshells crack”, which are consistent with the problem and implicit in it.

The LODE exercises rely on both such solving capabilities of a constraint programming system, namely, that of deciding on and that of deducing new Allen relations, consistent with a LODE story. First we need to present such exercises, then we can return to constraint programming by explaining its role in the architecture of LODE and in solving the exercises.

## Chapter 4

# LODE: Educational Exercises

LODE presents a list of e-stories the child can choose among. They are simplified versions of traditional children's stories, such as *The Ugly Duckling*, so that the language is more suitable to an 8-year old deaf child; they are also enriched with explicit temporal relations so as to concentrate the attention of the child on temporal reasoning. The child has to choose a story from the list in order to begin his/her work session. The chosen story is presented, split across different pages; there is one sentence with an explanatory image in each page. After reading the story, the child can start an exercise session for reasoning on the story. Uncommon words that may be difficult for deaf children are explained in a dictionary. In the remainder of this chapter, we describe the dictionary and the exercises of LODE.

### 4.1 Dictionary

The dictionary collects uncommon words that may be unknown to deaf children. The words are proposed on the screen together with an image explaining their meaning and a short textual explanation; example sentences are also available that help place and infer the meaning of a word in the appropriate context. The dictionary should be of assistance to children in the comprehension of the story and the association grapheme-meaning, a step which may be necessary with young deaf users. Children can consult the dictionary while reading the story or solving the exercises of LODE.

### 4.2 Global Reasoning Exercises

In LODE, we have two main types of reasoning exercises: comprehension exercises and production exercises.

### 4.2.1 Comprehension

In comprehension exercises, the child is presented with temporal relations connecting events of the story; the relations may be implicit in it. More precisely, the child is proposed four temporal relations. The child is asked to judge which relations are consistent with the text he/she has already read. The four cases can be constructed with the assistance of the constraint programming system that can determine which temporal relations are consistent with the story.

### 4.2.2 Production

The child is asked to tackle three main kinds of production exercises, explained in the following.

- P1. The child is shown an unordered sequence of temporal events extracted from the story; his/her task is to drag the events into the right temporal sequence, namely, one which is consistent with the story. The consistency is decided by the constraint programming system.
- P2. The child is shown scattered sentence units extracted from the given story; then he/she should compose a sentence with them, forming a temporal relation consistent with the story and which may be implicit in the story. For instance, suppose that the available sentence units are: before, while, after, the big eggshell cracks, the small eggshells crack. Two are the possible correct sentences the child can compose, consistent with the story. One is: the small eggshells crack before the big eggshell cracks. The other sentence is: the big eggshell cracks after the small eggshells crack. If the child composes a wrong sentence, because it is inconsistent with the story, LODE will suggest how to correct the sentence with the help of the constraint programming system.
- P3. The child can also reinvent the chosen e-story by using selected events of the e-story. More precisely, first LODE proposes a set of events extracted from the chosen e-story. Then the child re-creates his/her own story by reordering the events along the timeline, to the child's liking. Anytime, the children can check whether the temporal relations they introduced are consistent in their story, e.g., it is not the case that an event is simultaneously before and during another; they can also ask for the assistance of LODE, or better, of the constraint programming system of LODE in setting new temporal relations between events, consistent with their own story.

### 4.2.3 Final Remarks

LODE assists children in all the exercises, e.g., they are shown the events they may reason on. The difficulty of the exercises for reasoning increases with the portion of the story the child has to learn. Thus, first LODE proposes the simpler exercises: these relate two temporal events which occur in the same sentence in the story, temporally rich. Then LODE proposes the more challenging exercises, namely, those that require a deep global understanding of the story and the creation of global temporal relations: these exercises relate two temporal events that are distant in the story.



Moreover, note that the comprehension exercises aim at stimulating the deduction of global relations between events of the story; the production exercises demand this and something else, that is, to compose parts of the story text. Therefore, the production exercises require better reading comprehension capabilities.

Through all its exercises, LODE stimulates children to learn and reason on e-stories in an inductive and implicit manner.

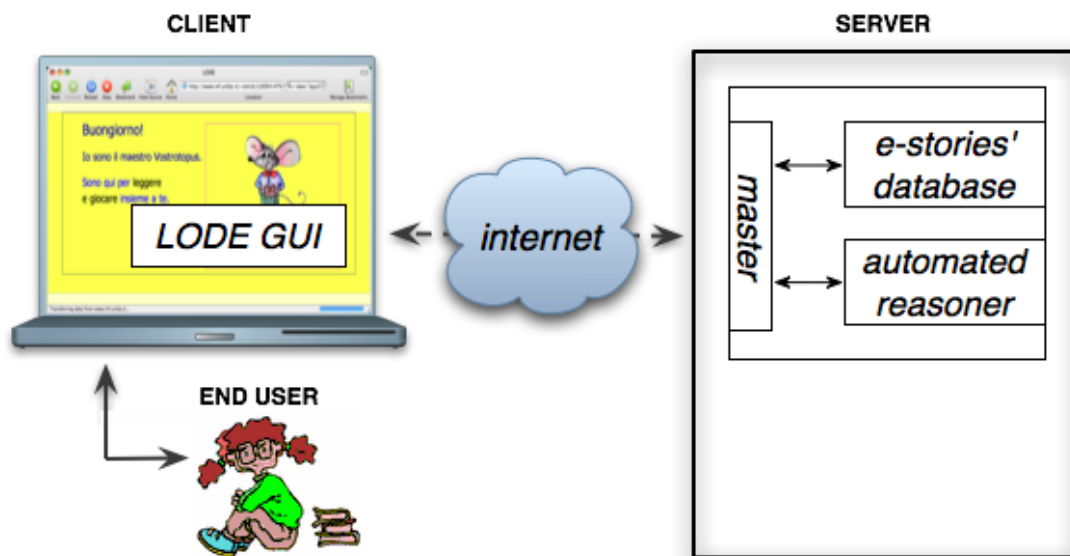


Figure 4.1: The client-server architecture of LODE.

## Chapter 5

# The Architecture of LODE

LODE has a web-based client-server architecture; see Fig. 4.1. We opt for this for several reasons. First, it makes LODE independent of any operating system. Second, it makes easier the updating of LODE; new features can be implemented without affecting the users, e.g., no need of installing new versions of LODE. Third, a web-based architecture promotes collaborative study: when they are on-line, the LODE users can work together and exchange their own stories or comments, whenever they feel like. Most importantly, they can use LODE at home, in a familiar and comfortable atmosphere, at their own pace, which can foster farther their learning process [27].

The client is a graphical user interface (GUI); see Fig. 7.1 for two screenshots. This is an AJAX application compatible with most web browsers. It works as the interface between the LODE user and the real system, the server, which runs on a remote machine. The GUI of LODE is described in more details in Section 7.

The server has a modular structure. The main modules are: 1) the *stories' database*, 2) the *constraint-based* module. See Fig. 4.1.

```
<EVENT class="E1">
  The small eggshells crack
</EVENT>
<EVENT class="E2">
  Mammy duck watches the big egg
</EVENT>
<TLINK event="E1" relatedToEvent="E2" relType="before OR meets"/>
<EVENT class="E3">
  The big eggshell cracks
</EVENT>
<TLINK event="E2" relatedToEvent="E3" relType="before"/>
```

Figure 5.1: A sample of a tagged story in LODE.

- 1) The stories' database is a simple repository structured as a file system. It contains temporally enriched versions of famous children's stories, in XHTML format. Events and relations are tagged in XHTML à la TimeML [28], the main difference being that the used Allen relations can be non-atomic; see Fig. 5.1 for an example.
- 2) The *constraint-based module* is composed of three main parts: a)  $ECL^{iPS^e}$ , the constraint (logic) programming system; b) the *knowledge base*, namely, an  $ECL^{iPS^e}$  program with the inverse and composition operations for the Allen relations; c) the *domain knowledge*, consisting of constraint problems modelling the temporal information of the e-stories in the database.

Although the constraint-based module was not integrated in the initial LODE prototype that was assessed with expert users, the module has been developed and tested for creating and checking some of the reasoning exercises; Section 6 details how this is done.

## Chapter 6

# The Constraint-based Automated Reasoner

Each e-story in the database can be modelled as a constraint problem in a semi-automated manner; the problem is included in the domain knowledge and then solved by the chosen constraint programming system. Let us make precise what we mean by modelling and solving in the context of LODE.

### 6.1 Constraint Modelling in LODE.

First we illustrate the main steps of the modelling with an example and then we generalise it.

*Example.* Hereby is part of the excerpt of the LODE story introduced in Section 3:

The small eggshells crack. Mammy  
duck watches the big egg. . . After some  
days, [...] the big eggshell cracks. (6.1)

The excerpt has 3 temporal events, tagged in the XHTML code as illustrated in Fig. 5.1:

- “the small eggshells crack”, classified as E1;
- “Mammy duck watches the big egg”, classified as E2;
- “the big eggshell cracks”, classified as E3;

The corresponding constraint problem, stored in the domain knowledge of of the constraint-based module, has variables

- E1E2 with domain equal to the set of the atomic Allen relations,
- E2E3 with domain equal to the set of the atomic Allen relations,

<pre> E1E2 &amp;:: [before,meets], E2E3 &amp;:: [before], allen_composition(E1E2, E2E3, E1E3) </pre>
--

Figure 6.1: The constraint model in  $ECL^iPS^e$  corresponding to the sample in Fig. 5.1.

- E1E3 with domain equal to the set of the atomic Allen relations,

and constraints of two main types:

- C1. unary constraints formalising the temporal relations of the excerpt (which are tagged with TLINK in the XHTML code in Fig. 5.1), that is: a constraint stating that E1E2 is **before** or **meets**; a constraint stating that E2E3 is **before**;
- C2. a ternary constraint on E1, E2 and E3 formalising the Allen composition operation; the ternary constraint on E1, E2 and E3 yields that **before** can hold between E1 and E3 since **before** can hold between E1 and E2, and between E2 and E3.

See Fig. 6.1 for the straightforward translation of the above constraints in the CLP language of  $ECL^iPS^e$ .

*Model.* As illustrated in the above example, here we adopt the same constraint model for qualitative temporal reasoning as in [26]. More precisely, the knowledge base contains

- the set  $A$  of atomic Allen relations, namely, the domain of each variable,
- a ternary relation, namely, `allen_composition`, for the composition of the Allen relations, e.g., `allen_composition(before, before, before)`.

Let  $E := \{E_1, \dots, E_n\}$  be the set of events tagged in a story of LODE. Then the domain knowledge of the constraint-based module contains the following:

- for each ordered pair of events  $E_i$  and  $E_j$  of  $E$ , a variable of the form  $E_iE_j$ ; their domain is equal to the set  $A$  of the atomic Allen relations;
- for each variable  $E_iE_j$ , a unary constraint  $C(E_iE_j)$  on  $E_iE_j$  which restricts the relations from  $A$  to those that are declared in the story (e.g., see item C1 above);
- for each triple  $E_i, E_j$  and  $E_k$  of events from  $E$ , a ternary constraint  $C(E_iE_j, E_jE_k, E_iE_k)$  stating that each relation in  $E_iE_k$  comes from the composition of  $E_iE_j$  and  $E_jE_k$  (e.g., see item C2 above).

## 6.2 Constraint Solving in LODE

LODE will employ both the following solving capabilities of a constraint programming system such as  $ECL^iPS^e$ :

- that of deciding on the consistency of a temporal constraint problem in the domain knowledge,
- that of deducing a/all the consistent Allen relation/relations between two events of a temporal constraint problem in the domain knowledge.

Such constraint solving capabilities can be employed in LODE in two main manners:

- to assist the LODE developers in the creation of the exercises; the constraint programming system allows us to automatically *deduce* the relations between events which are consistent with the e-story, such as *before* between E1 and E3 of (6.1);
- to assist children in the resolution of exercises. In the comprehension exercises and in some production exercises (see Section 4), the constraint programming system can be used to *decide* on the consistency of the relations proposed by children. Moreover, in the P3 production exercises, a child can recreate his/her own story, setting a different order among events of the story along the timeline; then he/she can transparently query the constraint system to *decide* on the consistency of the recreated story or to *deduce* a/the consistent relation/relations between the considered events.

## 6.3 Why Constraint Programming in LODE

In the following, we try to sum up the main reasons for adopting constraint programming in LODE; we split our summary in two parts, one for constraint modelling, the other for constraint solving.

### 6.3.1 Constraint modelling

- First of all, the constraint modelling is human readable as it is closer to the structure of the temporal reasoning problem; by way of contrast, a SAT encoding would be less manageable and readable for the modeller.
- The modelling can be semi-automated: events and explicit relations of an e-story are first tagged in XHTML; then a script can automatically read the XHTML tags and translate them into variables and constraints of the corresponding constraint problem.
- Last but not least, the constraint model for temporal reasoning that we adopted here allows us to employ a generic constraint programming system, such as  $ECL^iPS^e$ , as is. In other words, this constraint model does not demand to implement dedicated algorithms for temporal reasoning with Allen relations; for instance, with this model, we can exploit constraint propagation in the form of hyper-arc consistency which is already available in the *propia* library of  $ECL^iPS^e$ ; instead, with the constraint model described in [29], we should use and implement path consistency.

### 6.3.2 Constraint solving

- Due to the solving capabilities of the constraint programming system, the users of LODE will be able to create new temporally consistent e-stories out of the available ones: a user

of LODE can query the constraint programming system to decide on the consistency of the relations in his/her story, or to deduce new temporal relations that can be consistently added to his/her story.

- Using constraint programming spares the LODE developers manual work in the creation of the exercises, a task which is tedious and prone to human errors: instead of creating, manually, a series of exercises, e-stories are first modelled as constraint problems; then the relations of the reasoning exercises, introduced in Section 4, can be deduced using the constraint programming system.
- Moreover, the exercises created in this manner can be easily updated, new exercises can be introduced and solved on the fly using the constraint programming system.
- Although the efficiency in solving temporal problems is not a critical feature of LODE at present, it may become a critical issue in future and more ambitious versions of the tool; in this respect, the model and the different constraint propagation and search procedures already available in the chosen constraint programming system will be of pivotal importance.

## Chapter 7

# The GUI of LODE

The user interface of LODE consists of web pages narrating children’s stories and proposing exercises. Each page of a story contains a written sentence and an image illustrating the relevant temporal events of the sentence; see Fig. 7.1 for two screenshots.

In both the story narration and the exercises, text is visually predominant so that the child must concentrate on it and not on the image. The background and text colours are chosen carefully; colour is an important aspect of graphical interfaces for children, affecting perception and mood of the user [5]. Yellow, a colour that enhances concentration, is used for framing the page. Blue, inducing feelings of calmness and serenity, is used for the text background. White is used as font colour to enhance the contrast between text and background. Temporal events, the core of the LODE exercises, are in bright orange in order to draw the users’ attention. Abstract or uncommon words in the story are linked to the dictionary of LODE.

Images should support deaf children in visually representing, comprehending and memorising the relevant events, namely, those involved in temporal relations—the chance of learning is enhanced when both the verbal memory system and the visual memory system are involved [15]. However, images must be carefully chosen. They must be simple and unambiguous, that is, each image must clearly identify the represented event and avoid unessential information that may mislead deaf children.

Let us now turn to specific features of the design of the dictionary and the reasoning exercises.

Words in the dictionary are listed in alphabetical order. Each word entry contains, in the following order, the word, a short explanation with simpler words, an image. The order is not arbitrary: we emphasise textual information because LODE aims at focusing the attention of children on the verbal language.

The reasoning exercises are presented as entertaining games, not mandatory tasks for which children are evaluated; children can choose to repeat each exercise as often as they like, as well as to skip it.

In comprehension exercises, the child is presented with temporal relations connecting events of the story. The visualisation of the Allen relations is a critical feature still under evaluation. We propose two visualisation methods: a textual method and a graphical method.



In the textual visualisation two images representing story events are connected by four choice boxes, each identifying an atomic Allen relation. The advantages of this method are: it is easy to implement and compact in size. However, children must precisely understand the semantics of the writing of the Allen relations. For an example, the reader can refer to Fig. 7.2.

In the graphical visualisation, an atomic Allen relation is rendered by the spatial position of the images along the timeline. The advantage of this method is that the semantics of the Allen relations is more visual hence likely to be more intuitive for deaf children. However, this method may take too large a part of the navigation window. Moreover, the child may have to spend some time in order to get acquainted with it. For an example, the reader can refer to Fig. 7.3.



Figure 7.1: LODE: a screen-shot of a client session: at the top, the first page of LODE; at the bottom, a story page.



Figure 7.2: A comprehension exercise with the textual representation.

 *Cerca la risposta giusta!*

Gli animali della fattoria insultano il brutto anatroccolo **E SUBITO DOPO** il gatto e la gallina lo insultano.



Gli animali della fattoria insultano il brutto anatroccolo **E DOPO** il gatto e la gallina lo insultano.



 *Riprova. Sarai piu' fortunato!*

Figure 7.3: A comprehension exercise with the graphical representation.

## Chapter 8

# Related Work

As claimed in the opening of this paper, research in deaf studies and information technologies seems to mostly revolve around applications for sign languages, such as LIS; considerable less attention seems to be devoted to the development of e-learning tools for the literacy of deaf children in verbal languages. Such an impression is confirmed by our overview of this type of e-learning tools. We present the main ones related to LODE in the remainder of this chapter; we start with tools developed in Italian, as LODE is being developed in verbal Italian; we then examine other relevant projects and tools for the literacy of deaf people.

Table 8.1 offers a comparison of LODE with the reviewed tools for the literacy improvement of deaf children with respect to their educational goals and means. Table 8.2 compares the interfaces of the most advanced tools in terms of interface interaction.

### 8.1 Italian Tools

In Italy, three systems were developed in the '90s to tackle different aspects of the lexicon or grammar of verbal Italian: *Articoli* [30] aims at teaching Italian articles (e.g., gender agreement); *Carotino* [31] is an interactive tool for teaching simple Italian phrases; *Pro-Peanuts* [32] deals with the correct use of pronouns. Please, note that none of these tools were developed exclusively for deaf children. This is clear in a tool such as Carotino, where instructions for children presume a certain knowledge of written verbal Italian and do not focus on issues with verbal Italian which are specific to deaf children.

There are several tools that aim at teaching stories to deaf or hearing-impaired children. In order to facilitate the integration of a deaf girl into an Italian primary school, teachers and students of the school created *Fabulis* [33], a collection of famous stories for children narrated using text and images, based on gestures and LIS signs. Another application born at school is *Nuvolina* [34], the result of a project realised in a fourth class of an Italian primary school. Also this project aimed at integrating a deaf girl into the class. *Nuvolina* is a multimedia tale with contents in Italian, English and French, written and spoken. The version in verbal Italian is also presented in LIS by means of short videos.

In the area of bilingual tools, employing LIS and verbal Italian, we found *Gli Animali della Savana* [35]. This is a multimedia software based on text, images and videos, featuring an actor who translates the written text in LIS; assisted by a lion cartoon, the user navigates through a series of pages presenting the life of ten wild animals. A more recent and ambitious project is *Tell me a Dictionary* [36, 37], the purpose of which is to offer both deaf and hearing children an interactive instrument to discover and compare the lexicon of LIS and Italian. *Tell me a Dictionary* is a multimedia series of six DVDs plus book volumes [36].

We also found references to an e-learning tool developed in 1994, *Corso di Lettura* [38], which aims at improving the reading capabilities of hearing-impaired children. Alas, we could not find further information on the tool besides this.

## 8.2 Non-Italian Tools

CornerStones, FtL and SMILE are all advanced projects, developing tools usable by deaf children. We describe them in the following.

CornerStones [39] is a tool for teachers of early primary-school children who are deaf, or have visual learning capabilities and literacy problems. Academic experts in literacy and deafness, along with teachers of deaf students participated in its development. An essential element of Cornerstones is a story taken from the PBS's literacy series *Between the Lions*, complemented by versions of the story in American Sign Language and other visual-spatial systems for communicating with deaf children; word games are part of the tool. As LODE, Cornerstones adopts storytelling to enrich the vocabulary and world knowledge of children, and improve their reading comprehension; however there is no artificial intelligence in CornerStones, which is essentially made up of a collection of web pages. Its interface is highly entertaining and visual, with catching images; text is always accompanied by a reference image as in LODE. Overall the tool is well suited to children.

FtL [40] has not been developed for deaf or hard of hearing children, but this type of users has also been considered. FtL is a comprehensive computer-based reading program that has been designed to teach beginning and early readers to read with good comprehension. FtL consists of three integrated components: a Managed Learning Environment that tracks and displays student progress and manages an individual study plan for each student; Foundational Skills Reading Exercises, which teach and practice basic reading skills, such as alphabet knowledge and word decoding, providing the foundation for fluent reading; Interactive Books, which integrate human language and animation technologies in order to enable conversational interaction with a Virtual Tutor that teaches fluent reading and comprehension of text.

SMILE [41] has different educational goals than LODE and the tools mentioned so far: SMILE is not an application for improving the literacy of deaf children in a verbal or sign language; instead, it helps them learn mathematics and science concepts. However, SMILE is mentioned here due to its innovative interface for deaf children, which is highly entertaining and based on virtual reality.

ICICLE and MAS are two projects for deaf adults; although they are not meant for children, whereas LODE is, we report on them below because they share similar educational aims with LODE.

The primary goal of ICICLE [42, 43] is to employ natural language processing and generation to tutor deaf students on their written English. ICICLE's interaction with the user takes the form of a cycle of user input and system response. The cycle begins when a user submits a piece of writing to be reviewed by the system. The system then performs a syntactic analysis on this writing, determines its errors, and constructs a response in the form of tutorial feedback. This feedback is aimed towards making the student aware of the nature of the errors found in the writing and giving him or her the information needed to correct them. Similarly to LODE, ICICLE exploits findings and technologies of artificial intelligence, more precisely, natural language processing. However, the current ICICLE prototype is not usable by children, e.g., its interface is highly textual and not suited to children.

MÁS (Making Access Succeed for deaf and disabled students) [44] is a project that aims at improving the reading comprehension of deaf people. SIMICOLE 2002 is a tool developed within MÁS. The tool is made up of thirty texts related to ten themes, e.g., leisure and free time, divided along three difficulty levels. Spanish Sign Language is used as well. Each text is also accompanied by a series of exercises that aim at improving morphological and syntactic aspects of texts which are difficult for deaf people, in order to increase their lexical repertoire and help them in locating essential information. As in CornerStones, there is no artificial intelligence in the exercises of SIMICOLE 2002; the assistance of a human tutor is necessary for the feedback.

Table 8.1: Tools for literacy improvement: educational goals and adopted technologies

	<i>Content type</i>	<i>Use of sign language</i>	<i>Dialogue interface</i>	<i>Reading comprehension</i>	<i>Active feedback</i>	<i>Grammar analysis</i>	<i>Speech recognition</i>	<i>Automated reasoning</i>
Articoli	exercises	no	no	no	no	no	no	no
Carotino	exercises	no	no	no	no	no	no	no
Pro-Peanuts	cartoons	no	no	no	no	no	no	no
Corso di Lettura	exercises	no	no	yes	yes	no	no	no
Fabulis	children's stories	yes	no	yes	no	no	no	no
Nuvolina	a tale	yes	no	no	no	no	no	no
Gli animali della Savana	Exercises	yes	no	no	no	no	no	no
Tell me a Dictionary	cartoons	yes	no	no	no	no	no	no
ICICLE	user's input	yes	yes	no	yes	yes	yes	no
Corner Stones	children's stories	yes	no	no	no	no	no	no
FtL	interactive books	no	yes	yes	no	no	yes	no
SIMICOLE	thematic texts	yes	no	yes	no	no	no	no
LODE	children's stories	no	yes	yes	yes	no	no	yes



Table 8.2: GUI Aspects of four main applications

	<i>Look&amp;feel</i>	<i>Embodied agent</i>	<i>Navigability</i>	<i>On the web</i>	<i>Specific for children</i>
Tell me a Dictionary	virtual reality	no	plain	no	yes
ICICLE	flat text	to be done	plain for adults	no	no
FtL	unknown	yes	unknown	no	yes
SMILE	virtual reality	yes	generally plain with visibility system status problems	no	yes
LODE	AJAX	no	generally plain	yes	yes

## Chapter 9

# Assessment

After implementing a first prototype of LODE, we assessed it with its intended users, following the usability methods described in Section 2.3. First, we heuristically tested it with adult experts, namely: several LIS interpreters and teachers of deaf children, a logopaedist, a linguist expert of deaf studies, two cognitive psychologists expert of deaf studies for assessing the educational goal of LODE; five software engineers expert of web usability, one specialised in assistive technology, and a emotional design expert for testing the usability of the design of our prototype. We also tested it with two deaf children assisted by their parents and teacher. Children were observed while playing with LODE; their actions were recorded on a video and stored in a log file. Information on the children's level of deafness, literacy and acquaintance with internet browsing were gathered prior to the test session by interviewing their parents. With the assistance of their teacher, children filled in a questionnaire concerning LODE, in order to know whether they liked the tool, its games (that is, the exercises of LODE) and the story they had read.

The adult evaluators approved our e-tool and the implemented educational tasks. In particular, they agreed on the importance of stimulating deaf children to reason not on isolated episodes, but on global narratives. The two children testing LODE appreciated it: the older child (13 years old) read quickly the story and resolved the exercises. He did not have problems with the navigation, whereas the other one (8 years old) was not able to navigate without the help of her teacher, mainly due to the kid's poor reading capability.

The second evaluation phase is in progress; it aims at assessing the efficacy of LODE in stimulating children to globally reason on e-stories. The test is planned as a controlled experiment; deaf and hearing children will be involved in the test, so as to minimise the risk of misinterpreting possible deficiencies of the interface design. The third evaluation phase, which is planned for 2009, aims at delivering the final LODE prototype which fully integrates all the modules of the LODE server.

Table 9.1 presents a synthesis of our evaluation plan.

Table 9.1: Evaluation of LODE.

	<b>Phase 1</b>	<b>Phase 2</b>	<b>Phase 3</b>
<b>Users</b>	teachers for deaf children and LIS interpreters; a logopaedist; a linguist, expert of deaf studies; a cognitive psychologist, expert of deaf studies; software engineer, expert of usability tests; deaf children	c. 7 deaf children: c. 7 hearing children, all aged 8–13	c. 7 deaf children: c. 7 hearing children, all aged 8–13
<b>Evaluation goals</b>	analysis of the educational goal of LODE, that is, stimulating deaf children to reason on global temporal relations; test of GUI prototypes of LODE with children	assessment of the efficacy of LODE in stimulating children to globally reason with temporal relations; assessment of the GUI of LODE with children	usability tests of LODE
<b>The LODE prototype</b>	a first prototype, with the dictionary, comprehension exercises, (P1) and (P2) production exercises	a second prototype, with several e-stories and the majority of the exercises implemented	the final prototype
<b>Evaluation methods</b>	heuristic evaluation, observational evaluation, questionnaires for children and their teachers or parents	controlled experiment, questionnaires for children and their teachers or parents	controlled experiment, questionnaires for children and their teachers or parents
<b>Status</b>	concluded	in progress	to be done

## Chapter 10

# Conclusions

In this paper, we presented the design of our e-learning web tool for deaf children: LODE. The tool tackles problematic issues encountered in their written productions in a verbal language, related to deductive global reasoning with temporal relations; it narrates famous children's stories and challenges children to reason on them with the so-called reasoning exercises. We motivated why temporal reasoning on stories, that is, we explained its relevance for the literacy of children, and hence the importance of eliciting deaf children to reason on the temporal dimension of stories. Temporal reasoning in LODE is based on the Allen relation algebra; since the tool is for deaf children, such temporal relations are graphically represented, as explained in this paper and first in [45].

The reasoning exercises of LODE can be developed and solved with the use of an automated reasoner, namely, a constraint programming system [46]. In this paper, we detailed and motivated such a use of constraint programming in LODE, both in terms of the adopted constraint model for temporal reasoning problems and in terms of constraint solving. Albeit the constraint-based module was not integrated in our initial prototype of LODE, it has been designed, and already used for creating and checking some of the reasoning exercises.

This paper also reported on the preliminary evaluation of the initial prototype of LODE and outlined a long-term evaluation plan for it. The first evaluation did not give us statistically consistent information on the usability of LODE; obviously, we need to evaluate LODE with more deaf children, its end-users. However, due to the difficulties in involving deaf children and their parents in such activities (see Section 2.3), our preliminary assessment already suggested that we are on a good track. It ensured us on the need of an e-tool such as LODE for global reasoning on e-stories, in agreement with the literature findings examined in Section 2 of this paper. Moreover, the evaluation gave us indications on what to improve in terms of navigability in the exercises, and the interface of LODE in general. For instance, of the ten rules for user interface design prescribed by [47], the first prototype of LODE did not feature the visibility of the system status and recognition rather than recall rules. Future versions of LODE will also feature a translation of the dictionary words into Italian Sign Language (*Lingua Italiana dei Segni*, LIS) in order to facilitate their comprehension to LIS speakers.

We reviewed and compared with LODE several e-learning tools that address literacy issues for deaf people. According to our overview and to the best of our knowledge, LODE is the first

automated e-learning tool tackling literacy issues of deaf children which goes beyond the syntax and grammar of the verbal language of the country of origin; in fact, LODE aims at stimulating global deductive reasoning—in particular temporal reasoning—on narratives with automated feedback. This is a distinguishing feature of LODE.

Last but not least, the application of constraint programming in LODE goes well beyond temporal reasoning: we could extend LODE to other kinds of global deductive reasoning, such as causal reasoning, which are critical for deaf children and for which constraint-based deductive reasoning could be beneficial.

# Bibliography

- [1] P. Jordan, M. Makatchev, and U. Pappuswamy, “Understanding Complex Natural Language Explanations in Tutorial Applications,” in *Proc. of the 3rd Workshop on Scalable Natural Language Understanding*,. New York City: Association for Computational Linguistic, 2006, pp. 17–24.
- [2] R. Gennari and O. Mich. Lode demo. [Online]. Available: <http://www.inf.unibz.it/~gennari/index.php?page=lode>
- [3] UNESCO, “Consultation on Alternative Approaches for the Education of the Deaf,” Report. 1984.
- [4] B. Arfé and P. Boscolo, “Causal Coherence in Deaf and Hearing Students’ Written Narratives,” *Discourse Processes*, vol. 42, no. 3, pp. 271–300, 2006.
- [5] P. Paul, *Literacy and Deafness: the Development of Reading, Writing, and Literate Thought*. Allyn & Bacon, 1998.
- [6] R. Simone, *La Terza Fase. Forme del Sapere che Stiamo Perdendo*. Bari: Laterza, 2000.
- [7] P. Calliari and M. Degasperi, Eds., *I Bambini Pensano con le Storie*. IPRASE del Trentino, Provincia Autonoma di Trento, 2007.
- [8] D. Berry, “Requirements for Maintaining Web Access for Hearing-Impaired Individuals,” in *Proc. of the 3rd International Workshop on Web Site Evolution*. Florence, Italy: IEEE Computer Society, 2001.
- [9] NCAM. (2000) Design Guidelines for Electronic Publications, Multimedia and the Web. [Online]. Available: <http://ncam.wgbh.org/publications/adm/>
- [10] M. Marschark, “Cognitive Functioning in Deaf Adults and Children,” in *Oxford Handbook of Deaf Studies, Language, and Education*. New York: Oxford University Press, 2003, pp. 464–477.
- [11] I. Fajardo, J. Abascal, and J. Canas, “The Role of Working Memory and Long Term Memory in Deaf Users’ Hypertext Navigation: Review of Guidelines for Web Accessibility,” in *Proc. of the 8th ERCIM Workshop: User Interfaces for All*. Springer, 2004, pp. 320–325.
- [12] I. Fajardo, J. Canas, L. Salmeron, and J. Abascal, “Improving Deaf Users’ Accessibility in Hypertext Information Retrieval: Are Graphical Interfaces Useful for Them?” *Behaviour & Information Technology*, vol. 25, no. 6, pp. 455–467, November 2006.

- [13] J. Nielsen. Kids' Corner: Website Usability for Children. [Online]. Available: <http://www.useit.com/alertbox/20020414.html>
- [14] (2007) 230 Tips and Tricks for Better Usability Testing. [Online]. Available: <http://www.nngroup.com/reports/tips/usertest/>
- [15] N. Adamo-Villani, "A Virtual Learning Environment for Deaf Children: Design and Evaluation," *International Journal of Social Sciences*, vol. 2, no. 2, pp. 124–129, 2008.
- [16] S. Torres, I. Moreno-Torres, and R. Santana, "Quantitative and Qualitative Evaluation of Linguistic Input Support to a Prelingually Deaf Child With Cued Speech: A Case Study," *Journal of Deaf Studies and Deaf Education*, pp. 1–11, July 2006.
- [17] E. Efthimiou and E. Fotinea, "E-accessibility to Educational Content for The Deaf," *European Journal of Open, Distance and E-Learning*, 2006.
- [18] N. Israelite, J. Ower, and G. Goldstein, "Hard-of-Hearing Adolescents and Identity Construction: Influences of School Experiences, Peers, and Teachers," *Journal of Deaf Studies and Deaf Education*, vol. 7, no. 2, pp. 134–148, 2002.
- [19] J. Allen, "Mantaining Knowledge about Temporal Intervals," *Comm. of ACM*, vol. 26, no. 832–843, 1983.
- [20] K. Apt and M. Wallace, *Constraint Logic Programming using ECLiPSe*. Cambridge University Press, 2006.
- [21] B-Prolog. [Online]. Available: <http://www.probp.com/>
- [22] ECLiPSe. [Online]. Available: <http://eclipse-clp.org/>
- [23] GNU-Prolog. [Online]. Available: <http://www.gprolog.org/>
- [24] C. Bessiere, Ed., *Proc. of the 13th International Conference on Principles and Practice of Constraint Programming (CP2007)*. LNCS, 2007.
- [25] K. Apt, *Principles of Constraint Programming*. Cambridge University Press, 2003.
- [26] K. Apt and S. Brand, "Constraint-based Qualitative Simulation," in *Proc. of the 12th International Symposium on Temporal Representation and Reasoning (TIME'05)*, 2005, pp. 26–34.
- [27] R. Meighan, "Home-based Education Effectiveness Research and Some of its Implications," *Educational Review*, vol. 47, no. 3, pp. 275–287, 1995.
- [28] TimeML. [Online]. Available: <http://www.cs.brandeis.edu/~jamesp/arda/time/>
- [29] R. Gennari, "Temporal Reasoning and Constraint Programming: a Survey," *CWI Quarterly*, vol. 11, no. 2–3, 1998, Amsterdam.
- [30] Articoli. [Online]. Available: <http://www.anastasis.it/AMBIENTI/NodoCMS/CaricaPagina.asp?ID=36>
- [31] Carotino. [Online]. Available: <http://www.anastasis.it/AMBIENTI/NodoCMS/CaricaPagina.asp?ID=40>

- [32] (1998) Pro-peanuts. [Online]. Available: <http://www.ciscl.unisi.it/persone/chesi/laurea/str.htm>
- [33] Fabulis. [Online]. Available: <http://www.bonavitacola.net/fabulis/>
- [34] (1998) Nuvolina. [Online]. Available: [http://www.areato.org/noquadri/ausiliDinamici/AusDnm\\_01\\_Dettaglio.Asp?IDAUSILIO=229&IDSEZIONE=5&FORMATO=G&VETRINA=N](http://www.areato.org/noquadri/ausiliDinamici/AusDnm_01_Dettaglio.Asp?IDAUSILIO=229&IDSEZIONE=5&FORMATO=G&VETRINA=N)
- [35] (1994) Animali della Savana, Gli. [Online]. Available: [http://www.areato.org/noquadri/ausiliDinamici/AusDnm\\_00-Titolo.Asp?IDAUSILIO=106&FORMATO=G](http://www.areato.org/noquadri/ausiliDinamici/AusDnm_00-Titolo.Asp?IDAUSILIO=106&FORMATO=G)
- [36] E. Insolera, G. Militano, E. Radutzky, and A. Rossini, "Pilot Learning Strategies in Step with New Technologies: LIS and Italian in a Bilingual Multimedia Context "Tell me a Dictionary",," in *Proc. of the 2nd Workshop on the Representation and Processing of Sign Languages, LREC 2006*, C. Vettori, Ed., 2006.
- [37] (2005) Tell me a Dictionary. [Online]. Available: <http://www.lismedia.it/demo01/home.html>
- [38] Corso di Lettura. [Online]. Available: <http://www.anastasis.it/AMBIENTI/NodoCMS/CaricaPagina.asp?ID=35>
- [39] CornerStones. [Online]. Available: <http://ncam.wgbh.org/cornerstones/overview.html>
- [40] R. Cole, D. Massaro, B. Rundle, K. Shobaki, J. Wouters, M. Cohen, J. Beskow, P. Stone, P. Connors, A. Tarachow, and D. Solcher, "New Tools for Interactive Speech and Language Training: Using Animated Conversational Agents in the Classrooms of Profoundly Deaf Children," in *Proc. of the ESCA/SOCRATES Workshop on Method and Tool Innovations for Speech Science Education*, 1999. [Online]. Available: [citeseer.ist.psu.edu/cole99new.html](http://citeseer.ist.psu.edu/cole99new.html)
- [41] N. Adamo-Villani and K. Wright, "SMILE: an Immersive Learning Game for Deaf and Hearing Children," in *SIGGRAPH 2007*. San Diego, USA: ACM, August 2007.
- [42] Icycle. [Online]. Available: <http://www.eecis.udel.edu/research/icicle/>
- [43] L. Michaud, K. McCoy, and C. Pennington, "An Intelligent Tutoring System for Deaf Learners of Written English," in *Proc. of the 4th International ACM SIGCAPH Conference on Assistive Technologies (ASSETS 2000)*, 2000.
- [44] (2002) The MÀS Project. [Online]. Available: <http://acceso.uv.es/mas/index.html>
- [45] R. Gennari and O. Mich, "E-learning and Deaf Children: A Logic-based Web Tool," in *Postproc. of the 6th International Conference on Web-based Learning (ICWL 2007)*, ser. LNAI/LNCS. Springer, 2008.
- [46] —, "Constraint-based Reasoning for E-learning with LODE," in *Proc. of the 13th International Conference on Principles and Practice of Constraint Programming (CP2007)*, ser. LNAI/LNCS, C. Bessiere, Ed. Springer, 2007.
- [47] J. Nielsen. Heuristic Evaluation. [Online]. Available: <http://www.useit.com/papers/heuristic>