

# CATCH: A Program for Developing World Wide Web CALL Material

**Erik F. Tjong Kim Sang**  
Department of Linguistics  
University of Uppsala  
*erik.tjong@ling.uu.se*

## Abstract

The World Wide Web (WWW) is an interesting medium for CALL material because it solves the portability and reusability problems of recent CALL systems. However developing WWW CALL material is hard because of the required document encoding and programming knowledge and the large amount of encoding work that is necessary. This paper describes the program CATCH which can be used for overcoming these barriers. CATCH converts small descriptions of CALL exercises to working HTML files which contain JavaScript software. It allows CALL developers to develop WWW teaching material in an easier and faster way.

## 1 Introduction

Computer-Assisted Language Learning, in short CALL, is the research area which deals with the development and evaluation of computer software and hardware that is used in a language learning environment. The recently published book *Computer-Assisted Language Learning – Context and Conceptualization* by Michael Levy focuses on evaluating CALL systems and providing a high-level description for them. We are interested in obtaining answers to more basic questions: What educational methods can be used in CALL material? And what software can be used for developing CALL material?

This paper provides answers to these two questions. We will start with describing a basic collection of educational methods that are usable in CALL material. After that we will introduce our program CATCH<sup>1</sup> which supports the development of CALL lessons that make use of these educational methods.

We have chosen the World Wide Web (WWW) as the target platform for our CALL material. The advantage of developing CALL material for WWW is that it will become accessible and usable from almost all operating systems with standard software that is available on most computers these days. This solves portability problems and opens the possibility for using the CALL material longer than until the next software or hardware update<sup>2</sup>.

---

<sup>1</sup>The program CATCH is available at <http://stp.ling.uu.se/~erikt/catch/>

<sup>2</sup>Reusability of CALL material is a big problem according to [Lev97]. CALL material is often discarded at the next software or hardware update.

## 2 Educational methods in CALL

In this section we will present a basic collection of educational methods that can be used in CALL material. Presenting a complete collection is impossible and therefore we will concentrate on the methods that we consider most important. We will start with describing exercise formats and continue with help and feedback facilities. The section will be concluded with a general part in which some other educational methods will be described briefly.

### 2.1 Exercises

The ability of CALL software to check the language level of students and subsequently determine the course material they should be subjected to gives CALL an advantage over language learning by using books or audio-visual methods. Checking the language level of students takes place in an exercise environment. Two advantages of exercises with automatical answer checking are the possibility for testing students from one group on different topics simultaneously and the opportunity for discovering students which problems immediately without a large investment of time by a human teacher [Sci95]. It is hard to overestimate the importance of exercises for CALL material.

Exercises can be characterized by four features:

**Exercise features:**

Purpose: practice and test.

Task: translation, question answering and gap filling.

Format: text, speech and images.

Input format: text, speech and multiple-choice answers.

In our view exercises can be used for two purposes. One is for making the student practice with the material. In this case the performances of the students on the exercise do not have to be registered. Students may change their answers as many times as they want and attempt to use the feedback provided by the CALL software to get all answers right. It is also possible for using an exercise for determining the level of a student. In that case the exercise should be regarded as a test. The results of the students on the test exercises are important for the software and the teacher. The answers should be registered and the students should not be allowed to modify them.

There are three basic exercise tasks. The first one is translation. In a translation exercise the students have to translate words or phrases from one language to another. The second task is question answering. Here students have to answer questions, for example about a text they have read. The third exercise task is gap filling. Here students are presented with a text in which words or phrases have been removed. Their task is to fill in the deleted material.

There are three possible exercise formats. The first one is the most common one: presenting an exercise as text. It is also possible to present an exercise as spoken material. This is necessary for testing the listening capabilities of the students. However one should be aware of the fact that development, storage and retrieval of speech samples requires much more time and computer resources than doing the same with written material. The same is true for the third exercise format: images. Exercises can, of course, also make use of a combination of these three formats.

There are three formats of answer response from students. The most simple format is the multiple-choice answer. For every question the students are provided with a number of possible answers

and they have to choose the correct one. This makes dealing with the answers much easier for the CALL software since every possible answer is known by the developer of the material. If the student is allowed to respond to a question by inputting text then it is not possible to foresee every possible answer. Recognizing errors in answers becomes harder especially when the answer consists of a sequence of words rather than a single word. The third input format, speech, is even more challenging for the CALL developer than text input. Beside the answer handling problem, which is equally difficult as for text input, it also includes technical problems dealing with the recognition of the texts uttered by different students.

Developers of CALL material are interested in finding out which exercises will help increasing the language level of the students in the best way. A.G. Sciarone has provided some interesting answers to this question for the topic of text understanding [Sci95]. The language level of second language learning students can be increased by making them spend more time on reading written texts and listening to spoken material. Students will spend more time on these tasks if they perform tests with obligatory result levels regularly. Dictation exercises are the best method for testing listening capabilities and gap filling is the best exercise task for testing text understanding [Sci95]<sup>3</sup>.

## 2.2 Feedback and help

CALL software reacts to input of students by providing feedback. In order for the feedback to be useful the CALL system should be able to recognize and classify the input. This is one of the most challenging tasks for developers of CALL material. Too often language teachers that evaluate CALL programs complain that a student has given a correct answer to a question but that the computer has failed to recognize it [vdL97]. The problem here is that computer software is unable to understand human language.

A consideration of the position of CALL in language education is necessary here. CALL systems are not being used for replacing human teachers but as an extra tool for aiding their work. This means that CALL programs do not have to be able to evaluate every possible language learning task<sup>4</sup>. CALL software may concentrate on the tasks that it performs best and use input formats that are relatively easy to check.

If CALL programs allow only simple input of students like multiple-choice answers or single word responses then we can attempt to specify each relevant answer in advance. In that case it is possible for the CALL developer to define useful feedback for these predefined answers. The answer set will benefit from an evaluation of the answers given by the students while using the CALL material [Sar98]. It is therefore desirable that CALL software is able to store the unforeseen answers of students so that they be evaluated by its developers.

Apart from general help facilities about program usage CALL software needs special aids for students. The most important of these is a searchable online dictionary. Language students often encounter unknown words which they have to look up and it is an advantage if they can use the CALL system for doing that as well. A problem is that foreign language students may have different first languages and this means that the dictionary may need more than one source language.

---

<sup>3</sup>Sciarone has compared multiple-choice exercises, question answering exercises and gap filling exercises. His conclusion was that multiple-choice exercises allowed too much guessing, answers on questions are too hard to check for the available computational linguistics technology while gap filling exercises provide both good testing facilities and allow checking answers with the available technology [Sci95].

<sup>4</sup>An example of a language learning evaluation task which seems impossible for a CALL system with the present technology is testing the communicational capabilities of language learning students. This task requires a human teacher.

## 2.3 Other methods

With exercises, feedback and help facilities our overview of educational methods that are usable in CALL systems is by no means complete. Another important method is narrative text which can be either written or spoken. However texts need to be accompanied by exercises to make sure that the students have read and understood them. Equally important are explanations about grammatical features of the language that is to be learned. These can be incorporated in a general CALL lesson but they are at least as useful as a part of the feedback provided to the students. A third educational method which is offered by some commercial CALL programs are language games. We regard these as free-format practice exercises.

## 3 CATCH

In this section we describe our authorware program CATCH. We will start with a general description of the software. After this we will present the different parts of the CALL lesson description language and give an example exercise definition. We will conclude the section with experiences of CATCH users.

### 3.1 Background

The World Wide Web (WWW) is an interesting medium for CALL material because it solves the portability and reusability problems that comes with many of the current CALL applications. However there are some problems with this medium as well:

1. a lot of technical knowledge about document encoding and programming is necessary in order to be able to develop a working WWW CALL application<sup>5</sup>.
2. a large amount of encoding information is necessary for even the simplest exercise<sup>6</sup> and
3. because students might have a slow network connection to the material CALL developers are not able to apply freely all techniques they would like to use (they should be careful with using many images, animations and sound files).

These problems hinder the development of CALL WWW material. There is not much we can do about the third problem. However the first and the second problem are solvable. Our proposed solution for these two problems is to create a CALL development language which ables language teachers to define information necessary for CALL applications without having to worry about the peculiarities of WWW document encoding. An external program will be used for converting the specifications to a WWW document. Our program CATCH works exactly like that.

CATCH is a document conversion program written in the programming language Perl [WCS96]. It converts documents in a CALL encoding language to documents in the encoding language of WWW: HTML. The CALL encoding language consists of markup tags which define the type of text

---

<sup>5</sup>For example, [GSS96] reports that while many departments at the University of British Columbia were interested using WWW based course material in the early nineties both quantity and quality of this material was highest at the Computer Science Department.

<sup>6</sup>[Ahl98] reports that for a WWW file with an eleven-question gap exercise 89tags and software.

parts. The tags used in this language are SGML tags: they start with the < character and end with >. There are two variants of each markup tag: the <xxx> tag which opens context xxx and the </xxx> tag which closes context xxx. Examples of these tags are the start of lesson tag <lesson> and the end of test exercise tag </exercise>.

The users of CATCH can specify CALL material without having to worry about technical details like answer checking and WWW document encoding. The CALL format documents are both smaller and easier to understand than the corresponding HTML documents. It is possible to use CATCH without having any knowledge of HTML. However we have foreseen that CALL developers will want to use HTML features that are not included in the standard output format of CATCH. For that reason CATCH allows HTML tags to be present in its input file. The program will accept these tags and copy them to its output file. In this way it is possible for CATCH users to adapt the format of their CALL lessons to their own wishes.

## 3.2 Parts of the program

The present version of CATCH (2.0) recognizes fifteen CALL material encoding tags. The choice of these tags has been based on the theoretical outline presented in section 2. The largest part of the tags deals with exercises (9 tags). Apart from that there are some general structure tags (3) and some tags that deal with other educational methods (3). The general structure tags deal with defining the CALL lesson context, the headings and menus with pointers to the different parts in the CALL lesson. CATCH offers three other educational methods for the CALL material: texts, word lists and grammar explanations.

CATCH offers two exercise types: practice exercises and test exercises. Students can work with the practice exercises on a stand-alone computer. However for working with the test exercises they will need a computer with a network connection because answer checking will usually be performed by a central computer. Both exercises allow two input formats: multiple-choice input and text input. The decision about what exercise format and what exercise task to use has been left to the developers of the CALL material.

## 3.3 User experiences

The first version of CATCH<sup>7</sup> has been tested by second-year Language Engineering students at Uppsala University in a one-point CALL introduction course. The students have used the program for developing CALL material for Swedish and French. They were positive about the software. The students had some recommendations for additional functionality which have been used for extending the previous version of the program. The CALL material developed by the students has subsequently been published on WWW<sup>8</sup>. We have received quite some enthusiastic reactions on these lessons from its users and requests for developing more.

Currently the program is being used in a five-point CALL course for developing material for Spanish and Russian. In this course we are cooperating with a professional language teacher. We have received many interesting recommendations from both her [Sar98] and the students [Ah198]. Some of their recommendations for additional functionality, like practice exercises, have already been added to CATCH.

---

<sup>7</sup>The first version of CATCH was called call2html.

<sup>8</sup>The CALL material is available at <http://stp.ling.uu.se/call/>

## 4 Concluding remarks

The World Wide Web is an excellent medium for the development of CALL material. However the amount of technical knowledge and the amount of HTML coding that are required for creating CALL lessons may prevent language teachers from using WWW for developing their own teaching material. In this paper we have described a general tool called CATCH<sup>9</sup> which can be used to overcome this barrier. CATCH converts descriptions of CALL material written in an easy to understand encoding format to HTML files. By working in this way it is possible for people without technical knowledge to develop CALL material in a faster way than by creating HTML files from scratch.

## References

- [Ahl98] Viktoria Ahlbom. *Interactive Grammar Exercises in Russian*. Department of Linguistics, Uppsala University., 1998. (In Swedish, to appear).
- [GSS96] Murray W. Goldberg, Sasan Salari, and Paul Swoboda. World wide web course tool: An environment for building www-based courses. *Computer Networks and ISDN Systems*, 28, 1996. Also available on <http://homebrew.cs.ubc.ca/webct/papers/p29/>.
- [Lev97] Michael Levy. *Computer-Assisted Language Learning*. Claredon Press - Oxford, 1997. ISBN 0-19-823631-X.
- [Sar98] Kariné Åkerman Sarkisian. *A Pilot Project for Using CALL for Russian*. Department of Slavic Languages, Uppsala University., 1998. (In Swedish, to appear).
- [Sci95] A.G. Sciarone. *The Computer in Second Language Learning*. Boom Amsterdam, 1995. ISBN 90-5352-205-0. (In Dutch).
- [vdL97] Hanny van der Linden. personal communication, 1997.
- [WCS96] Larry Wall, Tom Christiansen, and Randal L. Schwartz. *Programming perl*. O'Reilly & Associates, Inc., 1996. ISBN 1-56592-149-6.

---

<sup>9</sup>The program CATCH is available at <http://stp.ling.uu.se/~erikt/catch/> . It runs on the platforms MSDOS, Windows and Unix provided that the programming language Perl is available.