# How Children Understand Concurrent Comics: Experiences from LOFI and HIFI Prototypes

Mikael Kindborg
*Department of Computer Science*
*Linköping University, Sweden*
*mikki@ida.liu.se, www.ida.liu.se/~mikki*

## Abstract

*In a study of how ten to eleven year old children understand program representations based on comic strips, it turned out that narrative interpretations were more common when using a low fidelity paper prototype than when using a high fidelity computer prototype. One explanation for this is that a computer prototype "sets the rules" to a much greater extent than a paper prototype, thus narrowing the set of plausible interpretations.*

## 1. Introduction

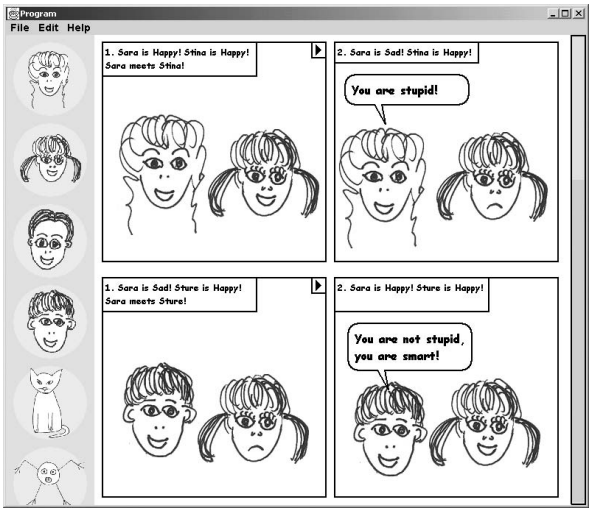
One approach for making programming easier is to reduce the mental distance between the program representation (the source code) and the resulting run-time program behaviour. By using a program representation that is similar to the run-time representation, the need for the programmer to do mental transformations is reduced.

Children often enjoy working with visual representations, both for fun and for learning. One representation that could be used to describe programs with animated objects and characters is the visual language of comics. Comics have a rich language for expressing dynamics and time, using strips of frames as the basic temporal device [4]. Since action is represented as static images it is possible to edit the actions and to get an overview of the representation. In addition, comics are popular and well known by both children and adults. Comics are also relatively universal and standardised.

The author suggested the use of comics as a program representation in 1987 [3]. Several systems based on graphical rewrite rules have been designed, for instance StageCast [6] and AgentSheets [1]. While graphical rewrite rules have similarities with comic strips, comics potentially have a much richer language for representing action, for instance by using various graphical markers such as speed lines. In the following a programming model based on comic strips is discussed.

## 2. Concurrent comics

The programming model studied represents programs as strips of frames. This is a screenshot from a computer prototype of this model:




The prototype has two windows, the program window (shown above) and the play area where the program runs. The program consists of strips that represent character behaviours. The first frame of a strip contains the precondition and the second frame contains the action. In the play area the user can move characters with the mouse, and when there is a precondition that matches the state in the play area, the action part is executed. The computation model is inspired by the concurrent constraint model used by ToonTalk [2]. Preconditions are evaluated concurrently using a non-deterministic policy. Thus the strips represent concurrent behaviours rather than a sequential story.

In the current prototype the possible preconditions for characters are character mood (picture), and minimum distance to another character. The possible actions are

Proceedings of the IEEE Symposia on Human-Centric Computing Languages and Environments (HCC'01)
0-7695-0474-4/01 $17.00 © 2001 IEEE

change mood (picture), move relative to another character, speak (using a voice balloon), delete character, and create character. These could be extended with other object types and operations, such as arithmetic ones.

## 3. Studies of LOFI and HIFI prototypes

The above programming model has been studied with ten to eleven year old children, using both low fidelity paper prototypes (LOFI) and high fidelity computer prototypes (HIFI). The following is a summary of these studies. The purpose was to investigate in which ways children understand and think about programming with comic strips. The studies were made in a small countryside school in a 4/5 grade class with 20 children. The children were given semi-structured programming tasks that aimed at making characters react when they meet. Each session was done with two children who worked with the researcher for about 40 minutes. The method used was qualitative and interpretative, asking the children questions about the program as they were creating it.

When working with LOFI prototypes the children had cut-out characters and voice balloons they placed in the frames. One sheet of paper contained the strips and represented the program. Another sheet represented the play area. For both prototypes an introduction was given to the children, explaining that in order to make something happen on the computer you must program it, and that nothing can happen besides what has been programmed. There were several differences between the children's understanding of the two types of prototypes. One such difference was the degree of narrative interpretation.

The children were asked what they thought would happen when two characters meet in the play area. When there was a strip with a precondition that matched the situation in the play area, the children answered with the action of that strip. However, when there was no corresponding strip, a common pattern was to come up with a new action. The following dialogue is a typical example:

RESEARCHER: What will happen when the boy meets with the girl?
CHILD: She says hello.
R: What in the program will make her say hello?
C: I don't know. Maybe she gets angry instead.
R: The only things you can choose from are one of the strips in the program or that nothing happens.
C: Perhaps they can become friends.

A narrative perspective like this one was very common when working with the paper prototypes. Another common pattern was related to the meaning of the order of the strips in the program. Many children associated the strip order with the order in which things would happen in the play area. When the researcher changed the strip order and asked how this would affect what happened, a common reply was that the action in the first strip would now happen first.

When working with the HIFI prototype narrative interpretations were less common and the order of strips was not considered as critical. This is an example of a situation similar to the one above, but when using the computer prototype. These children had not used the paper prototype, but the pattern was similar for children that used both prototypes.

R: What will happen when the boy meets her?
CHILD 1: She does not speak with him.
CHILD 2: We have not programmed him.

## 4. Discussion

One explanation for that the interpretation patterns differ between LOFI and HIFI prototypes is that a computer prototype "sets the rules" in a way that is not possible to simulate with a paper prototype. Children seem to assimilate and accommodate to the behaviour of the computer prototype, but when they use a paper prototype they invent things more freely and unrestrictedly. Another explanation is that the paper medium is close to printed comics, and thus associates to stories and narratives. The computer, on the other hand, might be associated with different attributes, like video games. The results found in this study are similar to the findings of Rader et al. [5] who reported on children taking a narrative perspective when using KidSim (now StageCast). In fact, the familiarity with comics might be a drawback for programming concurrent behaviours because of the association with stories. The next step in this research is field studies of programming in classroom learning situations.

## 5. References


[1] AgentSheets Web Site. http://www.agentsheets.com

[2] Kahn, Ken, "From Prolog and Zelda to ToonTalk", *Proceedings of the International Conference on Logic Programming 1999*, MIT Press, 1999.

[3] Kindborg, M and Kollerbaur A., "Graphical Tools for Description of Dynamic Models", *Proceedings of INTERACT'87*, Stuttgart, Germany, 1987.

[4] McCloud, Scott, Understanding Comics, HarperCollins Publishers, New York, 1993.

[5] Rader, C., Brand, C and Lewis, C., "Degrees of Comprehension: Children's Understanding of a Visual Programming Environment", *Proceedings of CHI'97*, Atlanta, 1997.

[6] StageCast Web Site. http://www.stagecast.com