Ludi: A Plug & Play Interface for Preschool Kids to Create Digital Games

Frank Hegel

Cognitive Interaction Technology | Center of Excellence (CIT-EC)
Sozirob Project, Bielefeld University

Abstract

This short paper presents the conceptual interface design of 'Ludi'. Ludi enables preschool children at the age of five to create their own digital games. By creating games with Ludi, children do not have to learn sophisticated instructions to create games because Ludi enables kids to create games easily by plug and play.

1 Introduction

Up to now, many electronic toys for preschool kids have been developed. On the one hand, most of the commercial toys are almost limited to follow a specific interactive path again and again. For instance, today's interactive books come with a kind of 'magic pen' that works like a mouse pointer. With such a magic pen kids can point the pen to a 'word' and the word will then be pronounced by an electronic speech production. However, the usage of such toys is limited to a specific interaction pattern. Consequently, playing with these kind of toys does probably not enhance the kid's creativity.

On the other hand, computers systems in general are less limited regarding creativity. Indeed, the most creative act to create games with computers is probably programming your own games. Subsequently, if someone is able to program a computer system in any language she will be able to program whatever she is able to realize – including computer games. Unfortunately, kids at the age of four to five are usually not able to program their own computer games. Additionally, parents and kindergartens understandably often dislike using computers to enhance the kid's creativity. This way, without learning any programming language, the conceptualization of Ludi enables kids to create their own digital content easily by plug and play.

After briefly outlining some inspiring related work in Section 2, the key aspects of the Ludi's conceptual design are introduced in Section 3. The following Section 4 presents an example how to create a classical game Pong by using Ludi. Finally, the last Section 5 concludes this article and gives an outlook to future work.

2 Related Work

According to Papert [1980], learning is focussing on how each child understands and creates its own reality. Tools are helpful in the sense that they are able to facilitate the understanding of facts. Consequently, he co-developed Logo, a programming language for educational use, where children among other things are able to control a virtual turtle to realize visual patterns. With regard to Logo, Papert found that children, for instance, create particularly an alternative understanding of geometry when creating visual patterns using Logo.

With regard to create own content by 'programming' another system of interest is the programmable robotics kit LEGO Mindstorms [e.g., Bagnall 2007]. This enables children to build their own interactive robots. Specifically, kids have to program the RCX-brick (Robotics Command System) by using visual programming languages like RCX-Code or ROBO-LAB which is based on LabView [see www.ni.com/academic/mindstorms]. With these easy to learn languages, school kids have the capabilities to control different actuators and sensorsthat are connected to the RCX brick. Kids are using these RCX bricks to build everything from robotic creatures to interactive kinetic sculptures, and, in the process, partially learning about engineering and design. Unfortunately, the use of Mindstorms is too complex for kids at the age of three or four, because they need to be literacy.

3 Concept

Basically, Ludi is a tool for preschool kids to create digital games. It consists of a central unit, several connectors, different input devices, output devices, and property rings (see Fig. 1). Input and output devices are paired with red rings by means of cables. This is important due to the fact that a cable visually indicates a physical connection between two entities.

Generally, Ludi's central unit has a display and five sockets evenly distributed around the central unit. Kids are able to activate the unit by plugging connectors into the sockets. Subsequently, altogether five connectors simultaneously can be plugged into the central unit.

Most important, each of the connectors carries a virtual object. From this it follows that if a connector is plugged into one of Ludi's sockets (see Fig. 2) a specific virtual object appears visually on Ludi's display. To illustrate, a virtual box within a connector appears immediately on the display as a box when a kid plugged the connector into a socket. Consequently, if there are five connectors plugged into Ludi's sockets, in total five virtual objects appear on the display.

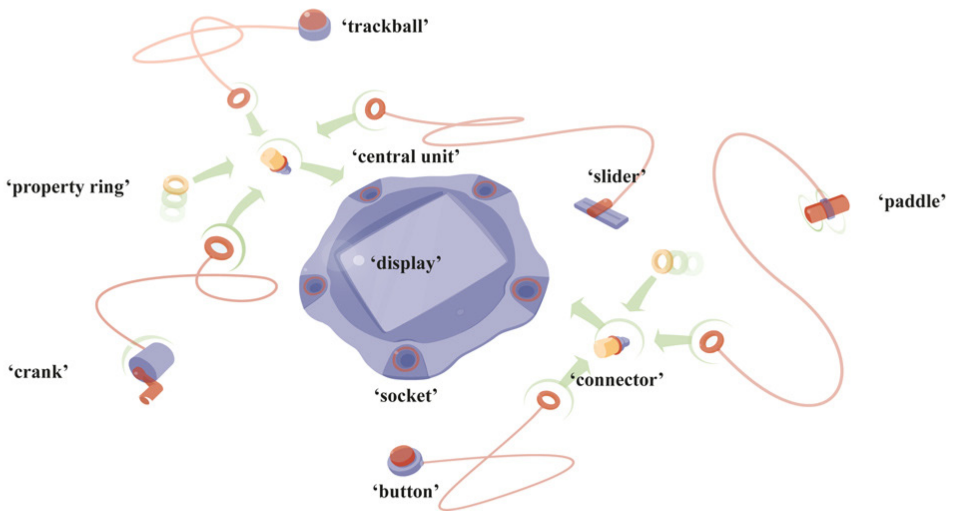

Figure 1: Schema of Ludi: Central unit, connectors, and input devices

Different objects like boxes, balls, flowers, triangles, magnets, suns, clouds were conceptualized to play a role when creating games with Ludi. Per default all the virtual objects are inanimate until a kid connects an input controller or a property ring to a connector in the central unit. For instance, an input controller with a crank (see Fig. 2, right) enables a kid to turn an object. A controller with a slider enables to move an object up and down or left and right. Importantly, only if an object is connected to an external controller it has a face (e.g., see Fig. 4) in order to indicate that the specific object is controlled by a human.

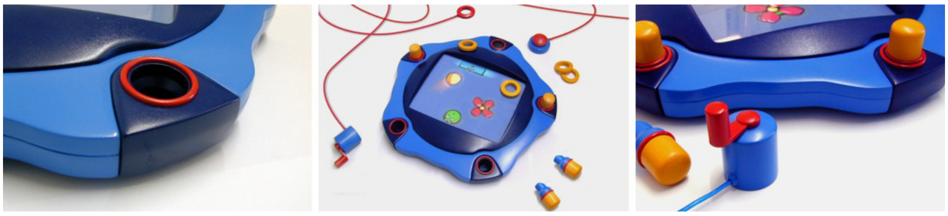

Figure 2: Product design of Ludi

Furthermore, property rings, for instance, possessing the property 'move' animate an object. Other property rings resize an object on the display or, for example, make the object virtual 'heavier'. Several additional properties to exchange speed, colors, and so on are conceivable, but not considered yet. Additionally, kids can only connect up to two entities (property rings, input or output devices) at maximum to one connector.

Importantly, all visible objects on the display relate to each other. This way, virtual objects have virtual physicalities interacting with each other. To substantiate, objects have virtual weights, materialities, surface conditions, intrinsic motivations, and much more. Consequently, each object can influence another by its specific properties: a ball bounces back when bumping into a box, a sun blends another object but will itself be covered by clouds, and so on. Therefore, manipulating virtual objects by using input devices has real consequences for all other visible objects. To illustrate, when using a crank to turn a flower as fast as possible the action accordingly produces wind and, hence, the light objects will be blown away to the display's edge. This is an important semiotic principle when designing plays or computer games. Something that does not indicate something else should not be there, because something that does not have any specific meaning results in incomprehensibility of the computer game [e.g., Salen & Zimmerman 2003]. The same is likely to be true for creating games with Ludi.

Due to the fact that creating games should not be limited to a display, output devices like propellers or external lights might enable the ability to transform output information of virtual objects to output devices. Altogether, by using input and output devices in combination kids can spatially extend their games.

To summarize, considering the conceptualization of Ludi, kids have the capabilities to create their games and to define their rules of play simply by manipulating objects by means of external devices and property rings. Thereby, developing ideas with Ludi is a kind of reflective conversation with objects of a specific situation.

The conceptualized system of rules in order to build digital games generally resembles the creative process of designing: seeing, moving, and seeing again [Schön & Wiggins 1992]. First, there principally is a situation. Second, a kid adds or removes objects, properties, or external devices. Third, due to the move a kid appreciates the action and sees the new situation. With such an appreciation the kid should be guided to understand Ludi's individual parts to create a whole game from its individual parts.

4 An Example: Creating Pong

Pong was the first commercial arcade video games developed by Atari in the early. It is a sports game that principally represents a virtual ping pong. The rules are simple: two players each control a paddle by moving it vertically across the left and right side of the screen. Players use paddles to hit a ball back and forth.

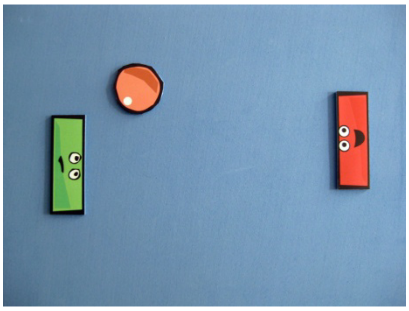

Figure 3: Paper prototpe of Pong created with Ludi

In order to explain how to create games with Ludi, Pong is a prototypical example due to its simplicity. To create Pong, kids only have to plug three connectors into the central unit's sockets – two boxes and one ball (see Fig. 3). In addition to the virtual objects, kids have to connect an input device like a slider to move each box up and down or left and right. Finally, the kids have to add a property ring to animate the ball. That is all kids have to do to create an abstracted and simplified version of the videogame Pong.

Moreover, there are several options to modify the classical game Pong easily (see Fig. 4). For instance, kids can replace a box by using a flower and thereby one player can blow the ball away instead of hitting the ball with a box. This way, also two kids can use flowers instead of boxes to play. To increase the level of difficulty, kids can also use two or even three animated balls instead of one. Another option is that one kid plays Pong on its own and uses a virtual magnet that influences extremely the ball's path. Obviously, there are many options to modify existing games like Pong and even to invent completely new games just by replacing objects or specific controllers – or by exchanging the object's properties. As mentioned above in the previous section, that is seeing a situation, modifying, and seeing again what kind of effect the modification has to improve a game.

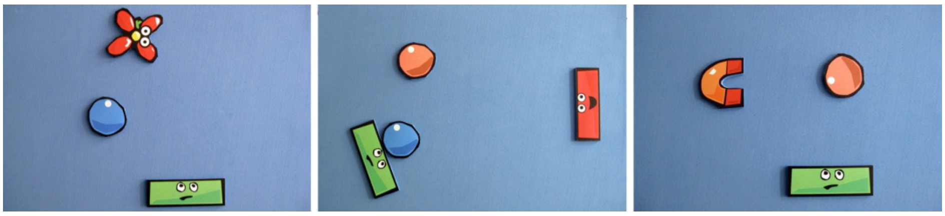

Figure 4: Ideas to simply modify the game Pong

5 Conclusion & Future Work

Up to now, Ludi is a design concept. This enables preschool kids to invent and realize their own digital games easily by plug and play. To create different games, kids simply have to insert up to five connectors into Ludi's central unit to activate virtual objects on Ludi's display. The virtual objects interact with each other and they can be controlled by external control devices. In addition, virtual objects can be manipulated by using different property rings to modify the virtual object's properties. To summarize, implementing digital games by using Ludi is combining virtual objects, external manipulators to control objects, and properties to modify objects.

Ludi is work in progress and regarding future work, the next steps are paper prototyping and an implementation of the concept. By using paper prototypes I want to interview preschool kids at the age of four to five what they expect such virtual objects in specific situations to do and how they would interact with the presented conceptual system. It is likely that the quality games invented with Ludi mainly depend on the object types and their relations to each other. Additionally, paper prototyping enables me to evaluate different screen designs and how many as well as what kind of virtual objects should be presented to motivate kids creating own games.

After a paper prototyping I am going to implement the concept to evaluate the principal ideas and the interaction with the system. As a result, I want to receive information whether preschool kids actually start or not to invent their own digital games by using Ludi. Moreover, an implementation and an initial usability test enables me to better understand the limitations of this conceptual design.

Acknowledgements

This work has been funded by the German Research Foundation within the Excellence Cluster Cognitive Interaction Technology (CITEC). Additionally, this work has partially been funded by the German Aerospace Center (DLR) with funds from the Federal Ministry of Economics and Technology on the basis of a decision of the German Bundestag under grant number 50 RA 1023.

References

Bagnall, B (2007). Maximum LEGO NXT: Building Robots with Java Brains. Variant Press, USA.

Papert, S. (1980). Mindstorms: Children, Computer and Powerful Ideas. Basic Books, New York, USA.

Salen, K. & Zimmerman, E. (2003). Rules of Play: Game Design Fundamentals. The MIT Press, USA.

Schön, D.A. & Wiggins, G. (1992). Kinds of seeing in designing. Journal of Creativity And Management 1(2), pp. 68-74.

Contact Information

Frank Hegel
Technical Faculty, Bielefeld University
Universitätsstraße 21–23
33615 Bielefeld (Germany)
E-Mail: fhegel@techfak.uni-bielefeld.de