CRITICAL FACTORS IN THE DESIGN OF PLAYFUL LEARNING ENVIRONMENTS[a]: REFLECTIONS ON “THE RETURN OF THE INCREDIBLE MACHINE: CONTRAPTIONS”Ô

Alvaro H GALVIS
The Concord Consortium, Concord, MA
alvaro@concord.org
http://www.concord.org/people

SUMMARY

This paper analyzes and makes explicit design principles that underline the software game *The Return of The Incredible Machine: Contraptions*,”[b]. In order to do this it takes into account two complementary perspectives – educational software engineering and theories of learning through play and games. One goal of the analysis is to identify the specific design features that make edutainment software successful, and, in doing so, to assist creators, evaluators, and consumers in making more informed decisions.

INTRODUCTION

It is not a school assignment – it is a free time activity that children, teenagers, and even adults love: solving and creating puzzles. In the program, *Contraptions*™, the puzzles are based on the behavior of devices in which the cause-and-effect principle is embedded. The logical integration of these devices results in machines that work! Users find this software exciting, challenging, and worth playing once, twice, and many times. I have seen this over the years with my own children and their friends, as well as with educators and parents: they spend hours and hours trying to solve or create functional puzzles. They apply their own intuitive knowledge about the physical behavior of things. They have fun while learning by experience, inquiry, trial and error, cooperation, or even by simple observation.

Many questions arise from this phenomenon: Why is playing this game so exciting for such a varied group of people? Why is it educational? How can we distinguish good “edutainment” from bad? This paper will try to provide answers to some of

[a] This document was prepared for “PLAYSPACE: An examination of learning in multicultural, digital play spaces”. This project was sponsored by a CILT/NSF seed grant. Further details on the project are available at http://concepts.concord.org/playspace

[b] Users must have access to any of the versions of *The Incredible Machine*, by Sierra Online. Its version 3.0, 1995, **TIM**™, **The Incredible Machine** runs on Windows 3.1, Windows 95 or Macintosh. In this article I will refer to this as TIM™. **The Return of The Incredible Machine: Contraptions**, by Sierra Online. Version 1.0, 2000, runs on Windows 95/98/2000 or Macintosh. In this article I will refer to this as *Contraptions*™

these questions, drawing upon the research literature and findings from the fields of educational software engineering and the theory of play.

EDUCATIONAL DESIGN IN *CONTRAPTIONS*Ô

To better understand this article, I invite the reader to find a copy of *Contraptions*™ and play with it – even better, to invite a child or a group of children to play with it For those who do not have this opportunity, or those who are interested in learning more about the educational design behind the software, I suggest turning first to the Appendix, and reviewing the major characteristics of this playspace, from the point of view of educational software engineering.

In addition I suggest the reader to go to http://www.nowthisisfun.com/Contraptions/ and read about this software. Professional reviewers of *Contraptions*™, the newest version of TIM™, report an 84% overall rating on their online rating system! The reviewer Mike McDermott, says “The legacy of the Incredible Machine continues with The Return of the Incredible Machine: Contraptions. This is easily the best puzzle game to come along in years and fans of the genre will not want to miss it. The multiplayer options could use a little tuning, but the improvement on the Contraption Builder as well as the 250 pre-set puzzles is sure to provide hours of fun for gamers of all ages” [1].

The important thing now is to identify why TIM™ is so exciting for such a varied audience. The enticing quality of this game is not just a matter of 3D graphics, sound and action. A sound design and implementation are also essential to capture such a varied audience time and again. Let us explore, then, what is behind this phenomenally successful software.

LEARNING FROM THE DESIGN OF *CONTRAPTIONS*Ô

Examining the structure of the game helps us to get a better understanding of two different groups of concepts: those specific to *Contraptions*™, its components and function, and the more general, related to overall design and/or evaluation of this kind of edutainment software. In the Appendix I have partially reconstructed the educational design behind this kind of software that combines play, modeling and simulation within the framework of a general microworld (in this case the planet in which we live). Many building tools (physical components with behavior) were used to define and to solve exciting puzzles under the control of the user.

Let’s now go deeper, and try to learn from this example what should be considered critical design principles for a playful learning environment. I will comment on two general categories of principles: those related to “playful” digital environments and those that deal with “educational” digital environments.

CREATING PLAYFUL DIGITAL ENVIRONMENTS

The following concepts concerning play, taken from Huizinga and Rieber will help illuminate the essence of the first challenge behind the design of a playful learning environment: that of making it playful.

> "Play is a free action, executed as such and experienced as something situated outside of normal life, but in spite of that, something that can completely absorb the player even without a material interest and without obtaining any benefit, which is executed within a determined time and place, which is developed within order and subject to rules and which gives birth to associations that tend to surround themselves with mystery or costumes for detaching oneself from the actual world. [2, p.26]"

> "Play is generally defined as having the following four attributes: 1) it is usually voluntary; 2) it is intrinsically motivating, that is, it is pleasurable for its own sake and is not dependent on external rewards; 3) it involves some level of active, often physical, engagement; and 4) it is distinct from other behavior by having a make-believe quality. [3, p.43]"

In the design of an interactive learning environment the following aspects should be considered in order to make it playful:

- control of the microworld by the user;
- contextual, constructive, and non-controlling feedback;
- adaptive and robust interface design;
- flexible, history-based sequencing;
- seductive induction;
- illuminating help on request; and last but not least important,
- fantastic tools and arguments.

Control of the microworld[c] by the user

When playing computer and electronic games, simulators or "sintonic[d] microworlds, many children enjoy the *sensation of control of the microworld* [4, 5, 6] – the pleasure of being able to think and make the ideas reality. This is something that people do not often experience, especially in a formal educational setting. The design of the microworld is critical to creating this pleasurable sense of control.

Contraptions™ uses an *intrinsic microworld* – consisting of a metaphor, argument, and objects - that are directly related to the concepts the designer wants the user to confront. Our living planet (or another planet with different gravity and pressure) is the frame of reference within which the Contraptions function, both those that the Professor has prepared and those that the user has built. This planet is also the environment where the challenges are to be solved. Intuitive physics and problem solving arise naturally within this microworld. The user is in *control of the microworld*, as long as s/he can:

- select or define the goal,
- keep or change the context variables,
- select and make discretionary use of the intervention variables (tools),
- use trial-and-error and reflection to pursue the goal, or
- use his/her imagination to define new goals and the tools to be used for achieving them.

However, not all the domains of knowledge offer the same potential for building environments in which one can have direct experiences related to the content to be learned. Because of this, *extrinsic microworlds* are superimposed on the concepts and skills to be learned, in order to make the process exciting. This layering of the fantasy structure is also common in some drill and practice programs, where the microworld is used as an external motivator and reinforcer of the action. For example, in games based on "hangman" or "hall-of-fame" motivational arguments, solving the exercises is the way to prevent your character from being hanged (in the

[c] *Microworlds* are subsets of the scenarios where it is possible to model and simulate the behavior of given worlds, real or imaginary, usually simplifying their functioning. Microworlds contain arguments, challenges, characters, and tools. A microworld offers the user the possibility of observing the performance of the modeled system when it is started, as well as to edit the model behind the world. *Intervention variables* (tools) are under control of the user, in the sense that s/he can decide if they are active and what their behavior should be. *Output variables* are tied to the expected results, usually have a graphic display showing the status of the system, but their values can also be displayed, depending on the wishes of the user. *Context variables* condition the performance of the microworld or its appearance to the user [11, 26]

[d] *Sintonic microworlds*, according to Papert [4] do not need to be learned, their metaphors are in the vital field of the learner, i.e., his/her previous experiences and expectations.

case of "hangman") or to get to first base (in the case of "hall-of-fame"). However, the knowledge behind these exercises is completely independent of the behavior of the microworld. The mental models one develops to solve the problems are not reflected in the microworld. Extrinsic microworlds do not provide users with the sensation of control over action; they simply provide rewards or punishments for achieving or failing to achieve an externally imposed task.

Contextualized, constructive, and non-controlled feedback

Intrinsic feedback, which is simply information about the task being performed [7], is essential in building playful, interactive learning environments. The sensation of control that we hear children talk about relates to this component. When users apply tools in the microworld to make changes, they must see these changes actually reflected in the microworld–sometimes in real time, sometimes when the user requests it.

Contraptions™ allows the user to explore each of the tools, getting intrinsic immediate feedback (s/he knows immediately if the tool fits somewhere, if the direction is appropriate, or if it can be joined or linked with others). It also allows users to assemble machines and, upon request (by pressing the START button), get intrinsic feedback about what works and what does not.

The feedback can be extrinsic also, becoming a reward that is external to the task at hand [7]. Extrinsic feedback is usually linked to achievement of the goal. In order to provide this kind of feedback, the system needs to know what the goal is, what variables reflect the desired status of the system, and the expected levels of performance. When the goal is achieved, the program confirms to the user that his/her solution corresponds to the expected one, and/or provides the prize that was announced when the puzzle or set of problems was launched.

Contraptions™ also makes use of external feedback. For example, when users solve a puzzle, the system tells them that they have achieved the goal, and allows them to see the Professor's solution, which is not necessarily the best. Similarly, when they define a puzzle, the system asks them to define the variables that signal when a goal has been achieved and the corresponding expected values. Having defined this, the system requires them to test that the system works. When they successfully do this, they get a reward message. In using external feedback related to puzzles with multiple solutions the system can assess the achievement of the goal, but it cannot determine whether a given solution is the best. So the external feedback, to be constructive, needs to encourage the user to find out if there is a better, simpler, or more efficient solution, instead of simply reinforcing the result obtained. Also, it is good to invite the user to explain his/her solution to others and to contrast it with someone else's solution.

Another critical point concerning feedback when playing games or solving puzzles deals with the "I give up" or "I've got it" behaviors of the users. When the user decides to give up, the system should not provide the expected answer. The system's answer should only be provided to the user when s/he has gotten his/her own answer, and only for the purpose of contrasting and testing other solutions to the puzzle, not as a means of giving the best answer. Providing answers when the users "give up" fosters dependency on the system and favors mental laziness. This is what happens in systems where, after two or three trials, the program gives the solution.

A final critical issue concerning feedback is that which Bos [7] calls "controlling feedback." This type of feedback inhibits reflection and does not promote thinking. An example of this occurs in arcade games, where users simply react to the event; they do not plan a solution or evaluate the status of the system and possible alternate solutions. Though the sensation of control one feels when playing these arcade games may be very high, to be constructively educational these games should allow reflection and not only reaction.

Adaptive and powerful interfaces

The human-computer interface is not just the communication system between the computer and the user [8]. The human-computer interface also makes the user's interaction with the program easy/difficult, boring/fun, attractive/disgusting, etc. Providing the ability for users to customize many of the features so that they are comfortable while working in the program is not an easy task. "Perception is relative" is a well-stated Gestalt principle [9]; as noted in the popular saying, "there is no disputing about tastes and colors." Features such as adjustable color, sound, music, and background objects are, therefore, important. Users feel comfortable and they appreciate the adaptability of the system when they are given control of some of these variables.

Prensky [10, p.46] suggests that members of the "Game Generation" are native speakers of the digital language of computers, video games, and the Internet. But these are not the only users of a playful learning environment. As Prensky says, there are also digital "immigrants" whose analog ways betray them in such behaviors as turning second to the Internet for information rather than first, or reading the manual for a new software rather than relying on the program itself to teach them how to use it. Both types of users will find the interface powerful, and it will allow them to approach the system in their own preferred way: from direct experience, trial and error, reflection, or learning beforehand how to use each of the different available functions. Finally, different users' abilities or disabilities should not be an obstacle for using a playful learning environment. If the user is hearing impaired, s/he should be able to read what is otherwise spoken. However, if s/he is illiterate, it should also be possible to listen to the messages.

Contraptions™ is a masterpiece in these aspects. Part of the fun of using the game is that users can adjust the interface and the way they use the system to their own preferences or needs.

Flexible history-based sequence

It is not just a matter of having control over the sequence of the action that leads the designer to keep track of past performance. Controlling the sequence of the action can be used, in effect, to help the user follow a suggested path of learning experiences. However, when the sequence is flexible, when the user is allowed to follow suggestions or to make his/her own decisions concerning what to do next without penalty or warning, this becomes a powerful feature. Combining menu-driven with history-driven sequences gives the user control over the sequence of learning activities without losing the knowledge derived from past experience [11].

In the case of *Contraptions*™, flexible, history-based sequencing is fully implemented. It allows different users to approach different types of puzzles in their own ways, based on their level of problem solving skills, topic preferences, the nature of the arguments, or the type of activity (e.g., create or solve puzzles) they want to perform. This adds to the appeal for users.

Seductive induction

Demos offer a good way to capture attention. When users know from the very beginning the kind of excitement that the system can provide, their internal motivation grows and expectations are raised very easily. If they also find that trying things out, taking risks, and making mistakes is not a punishable activity but a very acceptable one, then they will be more comfortable exploring each of the new devices or tools that are available. However, if they want to be sure before committing themselves, it is useful to try out an interactive demo that shows what they can do without losing the control of the device being explored.

This kind of seductive induction is an easy way of learning about each one of the components of the system, as it allows users to learn by experience but also by demonstration. In *Contraptions*™ the user has the means of learning about each one of the devices just by trying it or by pressing the magnifying glass and getting an interactive demo.

Illuminating help on request

Illuminating with indirect light [12] is a wonderful principle for helping people learn without imposing the instructor’s mental model about the content to be learned. In problem solving this is a particularly difficult teaching challenge: The designer does not want the user to be discouraged because of the level of difficulty or obscure nature of the problem. According to Vigotsky [13] problems ought to be in the proximity zone of the learner if they are to be relevant. At the same time, the system is not supposed to be so helpful that asking for help becomes the solution. It is good to give the user control of the kind of help s/he wants (i.e., hints or no hints, guidance without giving the solution, demos about the functioning of the parts of the system, guided interactive tours). This places the possibility of learning in the hands of the learner. *Contraptions*™ is a wonderful example of this.

Imaginative tools and situations

What makes an environment creative and fun is not the power of the tools available within it. Realizing the power behind a productivity tool such as a word processor or a spreadsheet does not mean they fun by themselves. In contrast, the tools in *Contraptions*™ are both powerful and fun. Each element is interesting in itself; it has properties that one can explore and use to change its behavior depending on the circumstances. Tools in *Contraptions*™ are surrounded by fantastic situations and exciting challenges (when solving problems); they provide the means for building new fantasies and challenges (when creating new puzzles).

If designers provide the user with provocative tools and arguments for building or solving exciting challenges, they are opening a gate to creativity, curiosity, and internal motivation. Moreover, they are providing powerful tools to build and test new mental models.

CREATING EDUCATIONAL DIGITAL ENVIRONMENTS

Educational digital environments are not synonymous with didactic, lecture-type digital presentations. What children and adults associate with high-quality computer software and Internet services are highly interactive environments. In these environments multimedia information is flowing and the user is engaged in a dialog with the software, information service, or other people on the network. How do we make such an experience educational? More importantly, how can we make playful digital environments educational?

There is no single recipe, since there are many ways to educate and different learners have different educational needs. I will explore the following points:

- educational approach,
- characteristics and needs of the audience,
- educational intention,
- motivation and reinforcement,
- learning activities,
- integration with existing resources,
- integration with existing curricula.

Educational approach

When we examine a software application such as *Contraptions*™ we realize that it is more that entertainment. It is also educational and has an underlying educational philosophy. The authors of *Contraptions*™ certainly had in mind a mostly constructionist way of helping people get the essential knowledge. In the game, this constructionist approach is to create and solve problems in the context of a given microworld. But the environment also directly teaches some essential things, such as the functioning of the different parts of the system and the functioning of the whole system. For example, this is seen in the first 50 puzzles when the user asks for help in order to learn about the intrinsic properties of the different objects. It can also be seen when the user selects the component “How to build Contraptions” which results in a tutorial by default.

Both algorithmic and heuristic approaches for learning [12] are encouraged in *Contraptions*™. The latter is predominant, since most of the activities are under the control of the user who decides what to do (i.e., which puzzle to solve, which contraption to build) and how to do it (i.e, with which tools, with/without hints, with/without contrasting his/her solution with the Professor’s solution). This variety of treatments reflects an eclectic educational philosophy in which different types of goals[e] and different styles of learning[f] are supported through different kinds of learning activities. The literature on "Aptitude by Treatment Interaction" [14] suggests combining treatments, most of them under the control of the user, and some under the control of the system.

The critical point in the above eclectic combination of educational treatments is that it has to be both selective and consistent. The designer of an educational software cannot be blamed, without knowing the reason, for being either directive or permissive in helping others to learn, by being either a transmitter of knowledge or a facilitator of learning. It is most important to know why and when to use a particular educational approach, with what purposes, and with whom.

Characteristics and needs of the audience

Understanding the characteristics and needs of the audience greatly affects the kind of stimuli a designer provides, the arguments s/he uses, and the motivators and reinforcers that drive the action.

In the case of *Contraptions*™, there is no explicit definition of the audience, neither embedded in the software, nor in the accompanying documentation. However, in examining the interface, there is clear evidence that children and young adults were taken into account in its design and production: Color, sound, voice, graphics and animations have the look and feel that young people are accustomed to when playing with video games involving high performance and adjustable interfaces. However, looking at the different sets of puzzles, which are organized in five sets of 50 puzzles each, one realizes that the intended audience may be groups of children or teenagers, either working collaboratively among themselves or with their parents

[e] There may be “reproductive”/low-level objectives concerning the appropriation of the means of the system as well as “productive”/high-level objectives concerning the final goals of the system

[f] The most common Learning Style classification includes: Field-dependent learners, who need external locus of control, as well as of Field-independent learners, who follow their own internal locus of control

as facilitators. The more users advance in the solution of puzzles, the more complex are the contexts they need to understand in order to discover what is going on, what the challenge is, and how to solve it with the given tools. This may require cooperation between children or interaction with adults.

This is not to imply that the software was not also designed for adults. In fact, it is interesting to find that many adults find this game exciting and worth using, not only with their children, but also for themselves. Adjustable interfaces and selective puzzles help the different users of the system feel comfortable while using it, thus engaging their interest.

Educational intention

The educational intent of the software drives the action. A designer may be very constructivist, however, if the intent is to help users memorize a poem, constructivism may not be the most effective approach. On the other hand, a designer may be very behaviorist but if the purpose is to help people create their own solutions to puzzles, such an approach will not work either. The designer's intention and educational approach should be, and can be, coherent. If they are not, the designer then needs to decide how to match goals with a compatible educational approach. Being clear about the goals and subgoals is critical.

In the case of *Contraptions*™, this software is most certainly designed for developing creativity. This is apparent in its two complementary dimensions [15, 16, 17]: “being creative”[g] (e.g., inventing new puzzles) and “having creativity” (e.g., solving given puzzles). Intuitive physics is the domain in which this creativity is to be developed, but with the clear expectation that it will transfer across knowledge domains.

In both modes of using the system (defining or solving Contraptions) the user is totally in control. If users are solving existing problems, they can decide which problem to solve. The user decides how to solve it, using selected tools and choosing how to combine them. Time is not a restriction, nor is the number of trials. Solutions can be tested as many times as the user would like, but without learning the system’s solution before finding at least one of his/her own. If the user is building Contraptions, s/he has total freedom to select and combine parts.

A subgoal within the system is to discover how different machines or parts function under different physical conditions. This can be accomplished by playing with the machines. The system also demonstrates the different functions of each part when the user sets the appropriate conditions.

[g] According to Jaime Parra [15 cited in 16, p.159] *having creativity* is having the capacity to establish unusual relationships and associations, to solve problems, to identify diverse alternatives, development of analogical thinking, fluency, originality, etc. *Being creative* has to do with the sources of inspiration, the brilliancy of ideas, spontaneity, freedom of spirit, psychological security, motivation, familiar environments, humor, etc.

Motivation and reinforcement

What moves the audience to interact with the software is a critical educational aspect. The challenge is not only that children decide to use the system (this can be internally or externally motivated), but that they keep using it because they want to, (internal motivation). *Contraptions*™ employs a highly intrinsic motivation system, as well as implicit reinforcement. Though the system initially declares that the professor is seeking an assistant and will select a person who has solved all of the puzzles, this is not generally what motivates repeated participation. Users agree that the satisfaction of building interesting and challenging machines (implicit motivation) and/or of solving a given puzzle (implicit reinforcement) is what drives the action.

The other critical motivational factor is that the puzzles to be solved should be relevant challenges in the zone of proximal development [13] and vital field [18] of the learners: Challenges that are not interesting, too easy, or too difficult may not motivate the learners to try them. In *Contraptions*™ puzzles can be selected to accommodate the level of difficulty that the user desires, as well as the significant and relevant arguments and goals to be achieved.

The possibility of playing with the adjustable background, sound and colors, makes this a friendly and exciting system, but by themselves, these features are not the key motivational factors. Nevertheless, if they were absent they might discourage certain audiences from playing repeatedly.

Learning activities

The core of the interaction between the user and the software occurs through specific learning activities. Each of these has an intention and an approach. Each responds to a need (motivation) and learns whether that need has been successfully met. It is evident that each of the scenarios has a different educational intention, with particular ways of helping learners to be successful. If the system were to be used with a different intent, a facilitator would have to create supplementary learning activities. For instance, someone interested in using *Contraptions*™ to help teach general problem-solving principles would need to include activities focused on finding and generalizing meta-knowledge based on the experience [19, 20, 21].

Integration with existing resources

What does it mean to “integrate with existing resources?” Most likely the reader has the answer, but let me share my point. Depending on the educational intent of the person selecting the software, it will be necessary to make use, not only of the software, but also of other learning devices. And this must be done in the right way.

Contraptions™ was apparently not designed to be used in formal education settings, but rather in informal settings. Teachers recognize the educational value of this program; they recognize that it can be used just for fun while helping children engage in an exciting mental activity, tied to the challenges that they are required to solve or define. But through my work with educators and children on Ludomatica[h], a Colombian project focused on playful learning environments for children, I have come to realize that if one wishes to go beyond the immediate goal, one cannot limit the learning experience to the computer. Educators are required to integrate different functional resources so that students can learn how to learn from different means and to make the best possible use of each of these resources.

In Ludomatica for instance, classroom size varies from 20 to 40 students with heterogeneous knowledge and level of development. In these classrooms, which have only one or two computers, teachers need to find a way to help all of the students obtain knowledge, build knowledge, and share knowledge concerning the topics of interest. This can be done by having them work simultaneously in small groups, each of them using different learning resources, but all of them searching for answers to similar relevant problems. They can make use of manipulatives, software, Internet, magazines, books, encyclopedias, physical activities, artistic activities, and many other resources that are integrated into the project. In a sense, the same intention drives all of the different uses of media and the same pedagogy is behind their use.

If the teacher does not have a clear set of goals and an understanding of how to help the learners achieve those goals, the use of a playful digital learning environment will never be more than a way of keeping the children focused on a fun task. Integration of playful digital learning environments is not easy; it is even more difficult if the teacher intends to use it within the context of a coherent playful pedagogy! There is research that can support teachers who are attempting to do this [22, 23].

Integration with existing curricula

Both informal and formal education settings are important, since both contribute to lifelong learning. This is, in essence, the great challenge of education: to help learners become increasingly independent learners who are able to contribute and participate in society. We know that edutainment software can be effective in informal education settings and we all wonder if it can be properly and successfully used in formal education. I believe that it can.

I have had the opportunity to work with non-conventional educators who combine constructivist and problemic[i] [24] pedagogy. I wonder if this combination of methodologies could make the difference in the integration of edutainment with existing curricula. If the curricular approach is centered on problem-solving, success is more likely because this is the focus of edutainment. Learning units could be focused on problems to be solved. In this case, both edutainment software and many other media can be functionally integrated within playful learning

[h] See http://lidie.uniandes.edu.co/ludomatica

[i] *Problemic pedagogy* is focused on exploring, analyzing, building knowledge, verifying knowledge related to problems. It is problem focused rather that content centered.

environments, all of them helping to build the desired knowledge and skills. The fun of learning would be combined with the fun of playing! What a dream!

FINAL REMARKS

This project has been a very exciting adventure for me. I hope that this interest in playful learning environments becomes contagious. I invite the reader to reflect on the different design dimensions I have identified for assessing the value of edutainment software. I wonder if these dimensions may be a good reference for analyzing the educational design of many playspaces. Also consider if the pedagogical ideas derived from exploring educational uses of playspaces are worth implementing in different educational settings, both formal and informal. To successfully achieve learning through play is not an easy task, but it can be accomplished if facilitators make the deliberate effort to consider the different dimensions I have mentioned.

ACKNOWLEDGEMENTS

This paper was prepared as a result of the project *Playspace: an examination of learning in multicultural, digital playspaces*. Each of the participants was in charge of reviewing software or experiences dealing with playful learning environments. A set of documents emerged from this effort, as well as a rubric for assessing the design of digital playspaces [25]. I want to thank my colleagues in this encouraging collaborative setting for their supportive and critical participation. Kallen Tsikalas, from Computers for Youth (NYC) merits a special recognition for her comments and proofreading of the document. On the other hand, Paul Horwitz and Carolyn Staudt from The Concord Consortium have helped me a lot reviewing and commenting early versions of the paper. Janet Hadingham has given the final touch by editing this paper. I wish to thank everyone involved.

APPENDIX: UNCOVERING THE EDUCATIONAL DESIGN OF *CONTRAPTIONS*ô

There are many ways to design educational software. At The University of Los Andes we have developed and documented a systematic way of approaching this design process [11, 26 27, 28]. This approach has enabled us to create highly interactive (as opposed to expository) educational software, and to describe and assess it[j].

The methodology we have developed [11] takes into account three dimensions of the design: educational, communicational, and computational. For the purpose of this paper, only the educational perspective is considered. In conceptualizing the educational aspects of a piece of software, we consider the scenarios and microworlds with which users interact. Specifically, we explore the different:

- arguments and challenges that drive the action;
- characters and tools (intervening and control variables) that can be used to change the status of the microworlds;
- contextual variables that can be used to change the settings of the system.

Scenarios[k] and Microworlds in ContraptionsÔ

The action takes places in a playful environment, *Contraptions*. This is the place where *The Professor* lives.

Posted at the entrance of *Contraptions* are different icons, providing a graphic menu that will allow you to decide what to do. Clicking on any of the signs results in an action. There is also a button labeled "press to hear" that will play an audio clip with a greeting from the Professor and will give you the basic thesis of the story line.

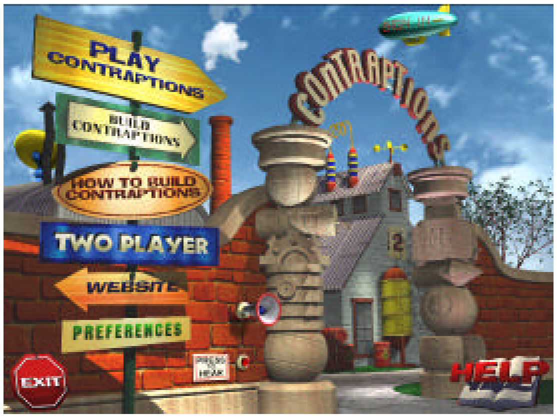


Fig. 1 Entrance to ContraptionsÔ

When you start the program you are given a demo of the kinds of things you can do, such as create a machine. This activity is initiated when a mouse grabs a piece of

[j] For instance, see in http://lidie.uniandes.edu.co/ludomatica/ciudadfantastica/menu.html the html presentation of the components of "Fantastic City" a highly interactive program that promotes creativity and problem solving.

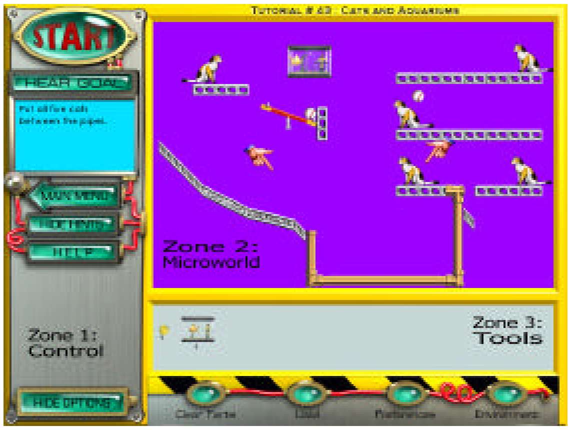
[k] *Scenarios* are the different virtual spaces where the user can interact. They are interconnected by logical gates that open depending on the choices that the users make. Each scenario has one or more communication zones that make a coherent interaction possible [11, 26].

cheese and this initiates the loading of all of the artifacts that you will be able to use in this game.

The first choice that you make will take you to the game console, where you are required to sign in. This console can be accessed at any time during game play, in case the user changes. There are different types of activities, depending on the sign that you choose. If you press "**Play Contraptions"** you will either be taken to the puzzle you were solving the last time you played with the system, or to the first one, if you are just beginning to play.

The first 50 puzzles are designed to help you learn about the characteristics of the different types of tools and options available in the system. To solve them you have to get information either by trial and error or by asking The Professor for help, about each available tool, what can be done with it, and how to use it. In addition to this initial set of 50 puzzles, there is a battery of 200 puzzles to be solved, ordered by increasing level of difficulty. You can customize the human-computer interface (sound, color, graphics) as well as the physical conditions that affect the puzzle (gravity and air pressure).

The following figure illustrates the three communication zones available at this level.



Put all five cats between the pipes.

Fig. 2 Communication zones

Zone 1, Control, allows you to start/ play your solution, to hear/see the goal, to change the activity, to play with/without hints, to call The Professor, and to get help about the objects you want. It also shows/hides the options below zone 3.

Zone 2, Microworld, is the zone where the action takes place when you press START. It contains the puzzle to be solved.

Zone 3, Tools, is the zone where objects and tools are available to be used to solve the puzzle.

If you press "**Build Contraptions**" at the entrance of *Contraptions*™, a workbench and tools are provided for you to use in building new machines. These can be tested, edited, saved, and recovered.

The structure and functionality of this level are identical to those in “Play Contraptions”. You are given a workbench with which you can define the title, goal, microworld, and tools to be used in creating and posting a new puzzle.

You can test your puzzle by selecting START. Pressing OPTIONS enables you to change the context of the puzzle (sound, music, graphics, color, gravity, air pressure) and to save it.

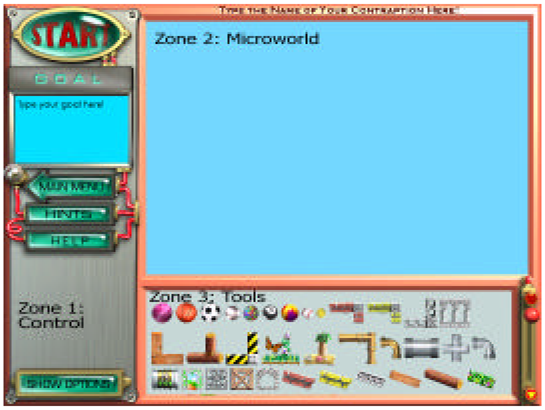


Fig 3. Build contraptions interface

If you do not know how to build puzzles, you can learn by pressing “**How to build Contraptions** ” in the central scene. This will give you a step-by-step tutorial, using The Professor as your guide.

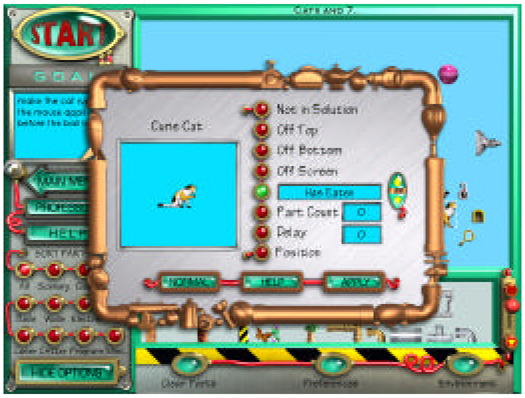


Fig 4. Tutorial on demand

The tutorial assumes that you already know how to play *Contraptions*™ and that you know the purpose of each of its tools and options.

Now you must learn how to define challenges and how to build microworlds using tools and contexts. Solutions are defined in the context of the behavior of different objects in the microworld. You can define hints, as well as test the system and save it with a name. By the end of the tutorial you will have created a puzzle.

You can use the system on your own or with a partner, in which case you can select the sign labeled “**Two players** ” in the main menu and enter the participants’ names, number of games to play, and length of turns. When pairs of users are playing the game, the system will manage the turns taken by each player and the length of each turn.

For a complete description of *Contraptions*™ select the arrow sign labeled “**Website**” and it will take you to http://www.nowthisisfun.com/Contraptions/ where a menu-driven text and graphic description of the system is given. This description provides details of the different tools that are available. This website could be used as a handbook describing the system, its requirements, opinions, and tips. From this site you can download new puzzles and releases of the program. For general help, outside the context of a specific puzzle, you may press the hypertext "**Help**" and get the information you want using a text menu.

When you are done with the system, click on the "**Exit**" traffic sign. The program will ask you to confirm, save the status of your games, and quit.

Argument and Challenge

When you select the door bell in the entrance of *Contraptions*™, the loud speaker says:

> "Welcome to *The Return of The Incredible Machine, Contraptions*. Hi! I'm The Professor. I am speaking to you through this loud speaker because one of my experiments went wrong. I'm currently radioactive and I do not want to contaminate you. I've assembled dozens of Contraptions for you to complete. If you can solve them all, I'll will know you are the one who has what it takes to beat my friends."

When you click on HELP in the entrance of *Contraptions*™, you will get:

> "Welcome to the *Return of The Incredible Machine: Contraptions*. Your host, The Professor is a world famous scientist. His inventions have made him extravagantly wealthy. For years he has worked behind closed doors, laboring away in solitude, deep within his top-secret complex.
>
> Now, he needs an apprentice…
>
> To assist him in picking a suitable apprentice, the Professor has invited you, and a select group of other scientists and inventors, to take part in a little contest. The rules of the contest are simple.
>
> A number of the Professor's favorite Contraptions have been left unattended in his complex. Your objective, besides having fun of course, is to solve all the Contraptions that the Professor has left behind. If you manage to do this, you'll be guaranteed a life of fame and fortune as the Professor's trusted apprentice."

Characters

The Professor and the active user (one of the two possible participants) are the only two characters involved in the scenarios. Neither of them is visible. You listen to the professor but you cannot see him. As the user you can "see" what is going on. You can move objects, click on them or on other available variables, but you are not represented by a character: the user IS the character behind the action.

Tools – Intervening and control variables

Available tools include over 100 different parts to mix and match. They belong to one of the following categories: scenery, gear, rope, balls, walls, electric, fire, laser, critter, programmable parts, and miscellaneous.

Using tools

Dragging a component from the tools zone to the microworld zone allows you to place the part where you want, while it is not interfering with other parts.

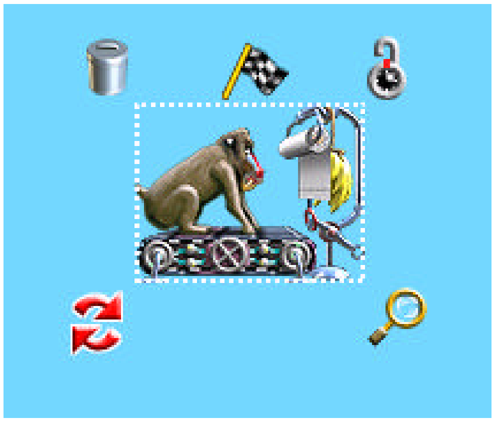


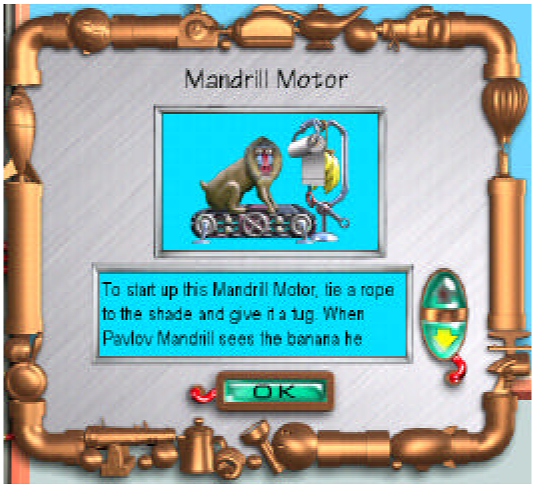
Fig. 5 Exploring parts

Moving the cursor to a given part in the microworld zone, allows the user

- to explore it (by pressing on the magnifying glass sign),
- to flip it horizontally (pressing on the red arrows sign),
- to determine whether the part is an output variable and, if so, its expected status (by pressing on the flag),
- to lock the part in a given place of the microworld zone (by pressing on the lock sign)
- or to return it to the tool's zone (by pressing on the garbage can sign).

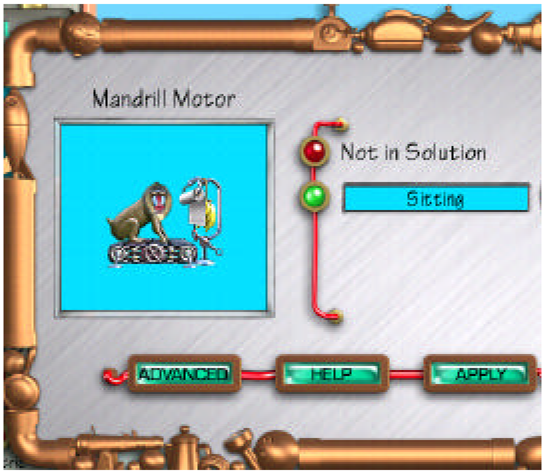
Understanding and defining desired status of tools

A given part can be explored by trial and error, by getting an explanation of its function (magnifying glass) and/or by a determination of the status (flag). The left figure below shows the window that appears when the magnifying glass is selected. The figure on the right below shows the window that appears when the flag is pressed. This flag window is also used to define the final status of the part, when it is to be in the solution of the puzzle.



To start up this Mandrill Motor, tie a rope to the shade and give it a tug. When Pavlov Mandrill sees the banana he

Fig. 6 Magnifying a part



Not in Solution

Sitting

Fig. 7 Defining initial and final stages

Contextual variables

There are two types of contextual variables: those that are in the context of the microworld and those that are in the context of the user, that is, the human-computer interface.

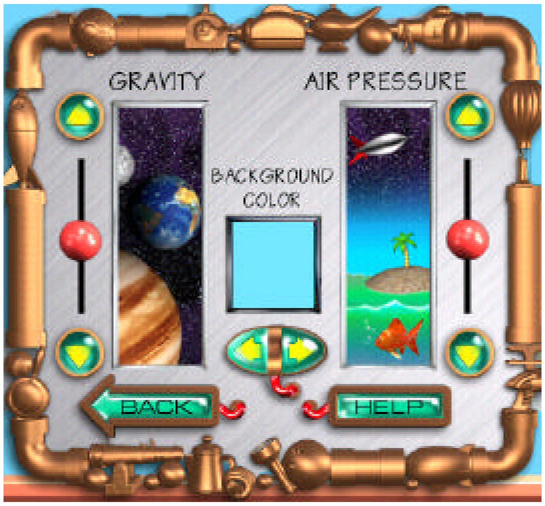


Fig. 8 Options

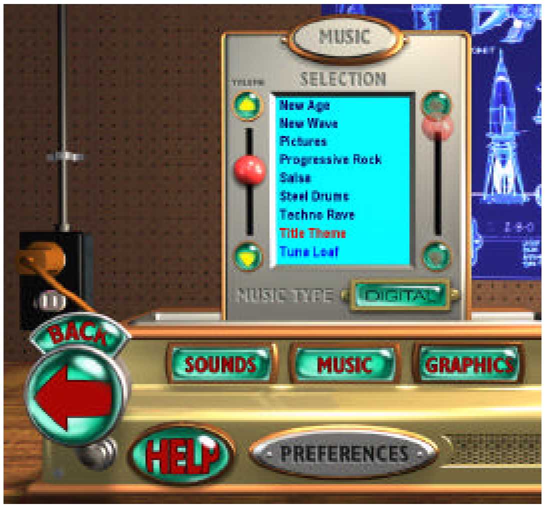


Fig. 9. Preferences

The above two figures show the windows that appear when you select the OPTIONS "Environment" (left side) and "preferences" (right side). You can play a puzzle in a "normal" environment, from the physical point of view (gravity and air pressure on the earth surface, an open space), or change it to another planet and pressure conditions. You can also customize the appearance of the screen, as well as the music, sounds and graphics to be displayed, in order to feel comfortable with the context of the game.

REFERENCES

1 McDERMOTT, M.(2000). *Game Over, Game Review* – The Return of The Incredible Machine: Contraptions. http://www.game -over.net/review/august2000/Contraptions/index.html

2 HUIZINGA, J. (1972). *Homo Ludens*. Buenos Aires: Emecé Editores S.A.

3 RIEBER, L.P. (1996). Seriously considering play: Designing interactive learning environments based on the blending of microworlds, simulations and games. *Educational Technology Research & Development*, 44 (2), 43-58.

4 PAPERT, S. (1980). *Mindstorms: Children, Computers and Powerful Ideas*. Brighton: Harvester Press.

5 GOLDEN, F. (1982). Here come the microkids. *Time Magazine*, May 3rd, 1982, 119 (18), 50-56.

6 DELVAL, J. (1985). Los usos de los computadores en la escuela. Madrid: Universidad Autónoma de Madrid, Facultad de Psicología (*mimeograph*)

7 BOS, N. (2001). What do game designers know about scaffolding? Borrowing SimCity design principles for education. CILT, PlaySpace Project, in http://concepts.concord.org/playspace (*PDF version*)

8 MORAN, T. (1980). A framework for studying human-computer interaction. In R.A. GUEDI *et al*, Methodology of Interaction, p. 293-302. Ámsterdam: North Holland.

9 WERTHEIMER, M. (1944). Gestalt Theory. *Social Research*, **11**, 78-99.

10 PRENSKY, M. (2001). *Digital Game-based Learning*. New York: McGrarw Hill.

11 GALVIS, AH, (1992). *Ingeniería de Software Educativo*. Bogota: Ediciones Uniandes.

12 DWYER, T. (1980). Heuristic strategies for using computers to enrich education. In RP TAYLOR (1980, editor), *The Computer in the School: Tutor, Tool, Tutee*. New York: Teacher's College Press, pp. 87-103.

13 VIGOTSKY, L.S. (1966). El papel del juego en el desarrollo del niño. En *Desarrollo de los procesos psicológicos superiores*. Barcelona: Editorial Grijalbo, p. 141-158.

14 SNOW, R.E and PETERSON, P.L. (1980). Recognizing Diferences in Student Aptitudes. In WJ McKEACHIE (editor), *Learning, Cognition and College Teaching, New Directions for Teaching and Learning*, Number 2, 1980, 1-24.

15 PARRA, J. (s.f). Preludio a la Inspiracion (*mimeo*)

16 ARCINIEGAS, M.E. (1995). Creatividad y maestros: Una alternativa para estimular la generación de la "sociedad de los poetas vivos". En República de Colombia, Presidencia de la República y Colciencias. *Fuentes Complementarias I – Creatividad, Formación e Investigación*. Colección Documentos de la Misión Ciencia. Educación y Desarrollo, tomo 5, pp.107-160.

17 GALVIS, AH (2000). Play, Puzzles and Creativity: Learning Engines for the Knowledge Society. Bogota: UNIANDES-LIDIE, Technical document ACE-01-00, Versión 2.0, March 2000 (*mimeo*)

18 WERTHEIMER, M. (1944). Gestalt Theory. *Social Research*, **11**, 78-99

19 MAYER, R.E. (1977). *Thinking and Problem Solving: An Introduction to Human Cognition and Learning*. Eugene, OR: University of Oregon Press.

20 WALLAS, G. (1926) *The Art of Thought*. New York: Harcourt

21 POLYA, G. (1957). *How to Solve it*. Garden City, NY: Doubleday

22 BEJARANO, G, *et al*., (1998), Conceptualización Pedagógica del Proyecto Ludomática. Documento conceptual PE-98-01, versión 5.0. Santafé de Bogotá, DC: UNIANDES-LIDIE, FURAPO, ICBF-PROTECCION, Proyecto Ludomática (*mimeografiado*).

23 TRECH, M. (2000). Encendiendo motores: Los acertijos y el juego en acción, a través de Guías Ludopedagógicas. Santa Fe de Bogotá, DC: UNIANDES – Proyecto *El acertijo y el juego como espacios de aprendizaje creativo*, documento técnico ACE-03-00, versión 3.0, Abril 2000 (*mimeografiado*).

24 BRAVO SALINAS, N. (1997). *Pedagogía problémica. Acerca de los nuevos paradigmas en educación*. Santafé de Bogotá: Secretaría Ejecutiva del Convenio Andrés Bello.

25 GALVIS, A.H., BOS, N., TSIKALAS, K., MOELLER, B., STAUDT, K. RUESS, K, SIGGINS, T. (2001). Rubric for assessing or designing digital playful learning spaces. In http://concepts.concord.org/playspace *(PDF Version)*.

26 GALVIS, AH (1997). Micromundos Lúdicos interactivos: Aspectos críticos en su diseño y desarrollo. *Revista Informática Educativa*, 10 (2), 191-204.

27 GÓMEZ, R, GALVIS, AH y MARIÑO, O. Ingeniería de software educativo con modelaje orientado por objetos: un medio para desarrollar micromundos interactivos. *Revista Informática Educativa*, 11 (1), 9-30.

28 MARIÑO, O, (1998). Desarrollo de micromundos educativos lúdicos– una perspectiva interdisciplinaria. *Revista Informática Educativa*, 11 (2), 193-200.