ENHANCING BOARD GAMES WITH ELECTRONICS

Daniel Eriksson, Johan Peitz, Staffan Björk

Interactive Institute
Game Studio
{daniel.eriksson|johan.peitz|staffan.bjork}@tii.se

1 ENHANCING BOARD GAMES WITH ELECTRONICS

This paper explores possibilities to enhance board games with electronics. Existing examples of augmented games are surveyed links between how various technologies could support specific tasks are explored. The main argument for this approach is to have a technology-focused view without the approach becoming technology-driven. Whether or not the tasks profit from enhancement with electronics is not discussed, since it mainly depends on the game.

2 BACKGROUND

The motivation behind enhancing board games is to make more immersive experiences for people in social situations. In computer games every player has his own display and the physical resources shared with the other players, if any, are spares. Board games are typically the opposite: players share one main display (the board) and physical resources are often exchanged. There are numerous examples of projects within this research area, either truly augmented board games or projects which share the basic premise (c.f. [Andersen04, Björk01, Invisibletrain, Ishii99, Lundgren02, Magerkurth04, Mandryk02]).

Even in games where every player has his own "display" it is still important that it is visible for the other players. There are however several games where the players have a private board and the gameplay focus on imperfect information, e.g. Poker and Battleboard, but in these cases there is usually also a public play area.

Although augmenting board games has a value in itself in that it can provide new gameplay possibilities, the showcase Socially Adaptable Games, within the EU funded project IPerG [IPerG], looks at board games because of their social qualities. In this project guidelines for how to make computer-augmented games socially adaptable through analytical studies and design experiments are developed. Board games do not only have social qualities because they bring people together for a period of time but also because they allow other social interaction to take place while they are played; that is, board games can generally be interrupted without disrupting the game. In contrast, if a player of a computer game is interrupted this can totally disrupt the game for that player, especially for games in real-time.

The Socially Adaptable Games showcase explores how pervasive games can be socially adaptable. This is done by studying how to augment board games for two reasons. First, as noted above, board games are more adaptable to changes in social environments than computer games. Second, the social environment of future pervasive games will have more in common with the environments in which board games are played than those in which computer games are played.

3 TASK ANALYSIS ON BOARD GAMES

For the purposes of this study, several board games (Settlers of Catan, El Grande, Citadels, Alhambra, and Ticket to Ride were analyzed using task analysis from the human-computer interaction field (see e.g. [Preece02] for a description of the technique) to identify game actions that can be augmented with electronics. The games were selected because of their relatively complex set of game mechanics. The method was used in a slightly different way than it is usually; instead of having the purpose of understanding how a general activity should be supported, the purpose was to play the game to analyze what the specific tasks were and what options were available to players.

3.1 Common tasks

Based upon a participatory study of gameplay in these games, as well as studying the printed game rules, tasks were identified that concerned manipulating the game state. This corresponds to identifying task related to operational rules instead of constituative and implicit rules in Salen and Zimmerman's classification of rules [Salen03]. These tasks were then analyzed to see if they could further be reduced into "atomic" tasks until the reduction could not be done further without being based upon measures outside the game state. The time span, the location of the actions, and the granularity of sensing are not considered since those parameters to a large extent depend on the characteristics of the game.

This method identified a high number of tasks, many of which do not necessarily improve gameplay, e.g. keeping track of players' scores, having too many tokens to sort and note-keeping of the game state. However, it could not be proven that these tasks affected gameplay negatively in all cases or that these tasks were representative of tasks that would be good to automate.

Many of the tasks seemed different when viewed first, but were basically the same when abstracting away game specific details, e.g. placing a card or a tile is in principle the same task.. To make it possible to relate tasks to technology that support or automate those tasks, and to do this so that these relations are applicable for new game designs, a general set of tasks was needed. This was created by generalizing all the atomic tasks depending on how players interface with the game state. This set was found to be able to describe user tasks on several levels of granularity and could be used as building blocks for a grammar for the interface between players and the game state.

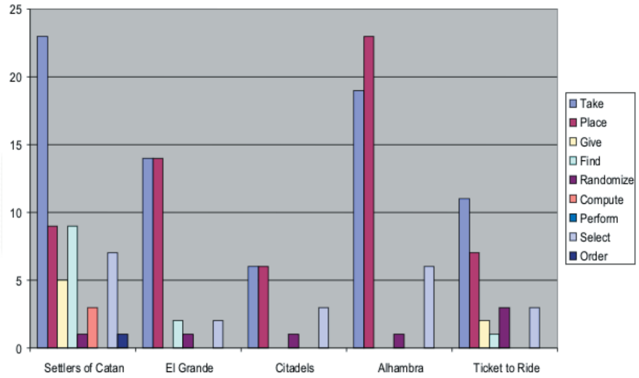

Figure 1. Task distribution for analyzed board games.

3.1.1 Take OBJECT from POSITION

To take an object has to be recorded to be able to save and load the game state. The object can be a tile, a card, a token or something else. It has a lot in common with placing and giving objects. These are the most common tasks in board games. Objects can either be taken from the game board, the box or from another players stack. It is only required to record the taking of objects from the game board for a minor copy of the game state.

Examples of these tasks are when a player: takes a resource token from the stack in Settlers of Catan, takes a card from the pile in El Grande, takes two gold coins from the bank in Citadels, takes a one money card from the four money cards on the table in Alhambra, or takes two train cards from the pile in Ticket to Ride.

3.1.2 Place OBJECTS on POSITION

To place an object is basically the opposite of taking an object. Examples of this task are when a player places a bought building on the board in Settlers of Catan or when a player moves a score marker in Ticket to Ride.

3.1.3 Give OBJECT to PLAYER/NON-PLAYER

To give or discard an object is similar to placing and taking objects, but it might not always be necessary to record which objects have been given. An example of this task is when a player gives a resource card to another player who has played the monopoly card in Settlers of Catan.

3.1.4 Find OBJECT

Finding objects on the board is a task that makes players search and identify on the board. This can be, for example, to search for ones own tokens or playable positions. A specific example from Settlers of Catan is when the players need to find the tiles that give resources after the dice have been rolled.

3.1.5 Perform SKILL BASED ACTION

Some tasks are not automatically successful or determined by rules. These are tasks where the success or failure is the outcome of a skill used or the way a skill is used as feedback to the system. Although both physical and mental skills could be covered by this task, it is primarily identified for physical tasks. “Compute EVALUATION FUNCTION” and “Select OPTION from SET OF OPTIONS” can be used to cover mental tasks. The task was identified during the creation of the generalized set without any specific examples from the games studied. However, examples of these tasks can be found in sports, pinball games and some board games, e.g. Jenga where the task of placing a brick on the tower is a gameplay skill.

3.1.6 Randomize

When something has to be randomized, like rolling dice or shuffling cards, this task is performed. Rolling the dice in Settlers of Catan and shuffling cards in El Grande are two examples of Randomize.

3.1.7 Compute EVALUATION FUNCTION

This task relates to the computation of equations and algorithms, but specifically not decision making. Counting eyes on the dice in Settlers of Catan is a trivial example of this while identifying possible build combinations in Ticket to Ride is a more complex example.

3.1.8 Select OPTION from SET OF OPTIONS

The tasks where players select options, e.g. where to place tokens. This is not the task of placing the token but of choosing where to place the token. It differs from “Compute EVALUATION FUNCTION” above in that it deals with decision making and there does not have to exist an explicit evaluation function nor does there have to be a quantifiable measure of how good the selection was at the point it was done. Examples include when a player has to select where to place caballeros in El Grande or decide where to place the first building in Settlers.

3.1.9 Order PLAYER to perform TASK

Players can be given orders to do something in the game, like discarding cards or giving objects to another player. A player orders the other players to give him all cards of a special resource after playing the monopoly card in Settlers of Catan. A general example of this task is when players have to remind other players that it is their turn.

4 LINKS

This section explores the possible links between the tasks found by task analysis and available electronics. Due to space restrictions, the technologies and examples are not described in detail here; the interested reader is referred to the IPerG deliverable Survey of Viable Technology for Augmented Board Games [Eriksson04]. Ties are made to projects that show how electronics are used for specific tasks.

4.1 Take OBJECT from POSITION

Things to consider when augmenting “Take OBJECT from POSITION” are how to know which object was taken and what position it had. Some of the answers to those questions will be answered here.

RFID can be used to detect the identity of the object. The object has to have an RFID-tag on it and be placed on an RFID-reader. The reader should preferably be able to identify multiple tags. In this case the position will not be known. There are however some kinds RFID-reader that detect both identity and position of the tag. iBrush uses an RFID-reader on the tip of an electronic toothbrush to detect which toothpaste is used; the tube of toothpaste had an RFID-tag on the lid [Bergsten03].

False Prophets [Mandryk02] uses an infrared light emitting diode to send out a unique code that’s detected by an infrared phototransistor. Since the board is made up of a grid with phototransistors, both the identity and the position are detected. CoverallSound [Åresund03] uses the relationship between two resistors to determine which patch is used and detect where the patch is placed, since there are a finite number of possible positions for placing patches. The patches are connected with the controller using conductive Velcro.

The position can be detected in several different ways, using resistive sensors. There can be Hall sensors used to detect presence [Andersson04], that change resistance when a magnet is close by, or LDRs (Light Dependent Resistors) that change resistance depending on the amount of light they are exposed to. If an LDR is covered by an object, this can be detected [Andersson04]. Phototransistors or photodiodes can also be used with the same setup to detect when an object is blocking the path. The objects can act as switches by being placed on buttons or have a bit of metal on the bottom and connect the input to ground. They can be placed on sensitive pressure sensors, but the weight might not be enough.

In a system where every player has a PDA, the items might not always require a unique physical representation. The objects can, if they only exist virtually, be transferred to player's possession either automatically or by entering a command.

4.2 Place OBJECTS on POSITION

To take and to place an object is very similar so the same things have to be considered for placing an object. The same technical solutions are useful with minor additions.

Ping Pong Plus uses several microphones to detect where on the table the ball hit. The position is triangulated from the input from four microphones on each side of the table [Ishii99].

Another way to augment the task is to give tangible feedback while placing. This can be accomplished with electromagnets and control them to attract if placement is possible and repel if not. This was used in a mock-up prototype to test the concept in Settlers of Catan.

4.3 Give OBJECT to PLAYER/NON-PLAYER

This is essentially a special case of the place task, but the position is of little interest and it has to be handled differently because the objects are given and not placed on a board. Every player can have an RFID-reader to keep track of what tokens the player has or a phototransistor to read the codes from an infrared LED. For some objects there is no real need for physical representation as they can be given electronically from one PDA to another via WLAN.

4.4 Find OBJECT

When a player is trying to find object he/she can be helped by a controller if it makes the sought after object(s) change appearance. Find OBJECT is either triggered as an event or by an action, e.g. a card being played. Visual and movement output devices can be used since they are easy to detect and locate, although visual output is the best.

Information can be shown on screen. In BattleBoard 3D a monitor is used to show the real status of the tokens [Andersen04], and KnightMage uses a large plasma monitor as a game board [Magerkurth04], while Pirates! [Björk01] and the Invisible Train [Invisibletrain] use the screen on PDAs to display information. A projector can be used either to project information on a surface [Magerkurth04] or to project directly on to the board as in False Prophet [Mandryk02]. In the last example the entire game board is displayed by a projector and changes could therefore easily be shown.

More subtle ways to display information are by using LEDs, lamps, thermochromic cloths or electro luminescent wires and plates. Elementary uses lamps to show information in another context [Callert03]. Electro luminescent wires are used in Pillows to display status [Pillows]. Interactive Tapestry uses thermochromic paint on cloths heated to alter their appearance [Andersson03].

Movement can be used to change the physical representation of a game board. Areas on the board can be heightened or lowered by using a linear servo. Vibrator motors can be connected to objects on the board, making them vibrate to indicate that the area or object is of interest.

4.5 Perform SKILL BASED ACTION

It is hard to provide generic suggestions for associating skill based tasks with electronics, since the augmentation depends upon which skill it is focusing on. There are however a couple of game genres dealing with skills. The skills can be reflexes, knowledge, multitasking capacity and strategy.

This category can be divided into two subcategories: actively assisting in the skill and detecting performance of skill based action. Examples of active assistance are sparse, a conceptual example could be to enhance the Labyrinth game by letting the game make the resistance of the knobs harder in a direction that would make the ball fall into a pit.

Performance interaction detection need not be limited to detecting success or failure but includes how a task is performed or at what speed. In iBrush it is of interest to detect at which frequency the teeth are brushed. This is accomplished by using an accelerometer to measure how often the toothbrush changes direction [Bergsten03]. GymAssistWear is a project where the motion of an exercise is supervised to make sure that it is done correctly. The motion is recorded from two accelerometers and compared to a predefined motion [Haglund03].

4.5.1 Novelty Games

Many examples of games based on this task can be found in Novelty Games, mechanical simulations of sports testing player's physical attributes and skills, commonly found in amusement parks before the 1930s. Besides inspiring the development of pinball machines, these games simulate racing, hunting, soccer (known as foosball), flying and building construction [Kent01]. The later and more complex games of this kind include light-sensitive targets that can be shot at with light guns (Seeburg Bear Gun) or use back-projected screens to provide an interactive background (Speedway). Kent sees these as the direct ancestors of video games [Kent01].

4.6 Randomize

This is a task easily performed by software by creating pseudorandom numbers with certain probabilities. It can then be used to remap the identity of objects, e.g. changing characteristics for a card with a certain RFID-tag. However, it is probably also one task that players often want to perform themselves. Performing data fusion on input from sensors or using a higher granularity than is easily perceivable can provide apparent randomness for players if the mapping between input and output is unknown or sufficiently complex. In many cases the tie to "Compute EVALUATION FUNCTION" is strong but the focus here is on how the result is generated rather than what a specific result is.

4.7 Compute EVALUATION FUNCTION

Computation can be non-trivial and based upon many parameters. In this case it is a task that has to be augmented with software. The result can be presented using any visual output and it can be automatically performed if it affects data on a PDA. One way of differentiating “Compute EVALUATION FUNCTION” from “Select OPTION from SET OF OPTIONS” is that the latter corresponds to the field of Artificial Intelligence.

If not all data is known, like the entire game state, the conditions have to be evaluated, or computed, before information can be presented. This is sometimes done before the task of selecting and might require input.

In other cases enhanced this task is often about reading data from the physical world and being able to process it. The data that has to be read can be the physical properties of an object e.g. which side that is up on a die or which part of the wheel is pointed at. Sensors that can be used to detect physical properties are for example:

For cameras and in some cases scanners the image can be processed by a computer to detect orientation of the object. Accelerometers detect the orientation of it self and has to be built into the object. The data has to be sent and preferably wirelessly. Special properties can be given to different positions of the object, as codes sent by IR LED, different resistance to measure the voltage over a resistor or different pins on the controller.

Two common ways of letting players perform the Randomize task by physical manipulation of objects is to throw dice or spin wheels. As a small design exercise, ways of making these measurable by a game system have been explored and are described below.

4.7.1 Augmented Die

Dice are often used in board games, even though they are only used in one of the games analyzed for this study. Here is a list of different possible solutions for reading the eyes of a die. Some are ideas for a crude prototype while others are more advanced, would take longer to implement and may not be cost efficient, at least in the near future.

- A very crude implementation would be to use a scanner as a dice table and a piezoelectric or electret microphone on the glass. The microphone would detect when dice have been rolled and start scanning them. The image can then quite easily be analyzed to count how many dice and calculate how many eyes face up.
- Instead of a scanner a camera can be used, but then the image processing would be more advanced since the dice on the image are harder to detect and it is harder to see which side is up.
- The eyes of the dice can be painted with a paint that reflects UV or IR light and have that kind of lamp underneath. A sensor, photodiode or phototransistor, can detect the amount of light that is reflected. The number of dice can be detected by weighing the dice table. This is not elementary.
- The sides of a die can have one RFID tag each and be played on an RFID reader. The reader should not be able to read the tags that are perpendicular to the reader so it should only be able to read the tag on the bottom of the die and the one on top of the die. This could probably be avoided by having an interfering core in the die. The core could be of metal and not be too large since it could interfere with the reading of both tags.
- A die can be augmented by having a PIC processor, a 3D accelerometer, a transmitter and a power source built in. The controller would read the accelerometer to detect when the die has been rolled and read the orientation. The transmitter would be awakened from sleep mode and the orientation sent to a receiver. As power source a battery can be used or if it is possible a power source that gets power from the motion of rolling the die.
- The die can also be built as in the last example but without the accelerometer. The controller could have an input for every side that is connected to a pull-up resistor and to a metal connector on the side of the die. Every side should have two contacts, one to the input and one to ground on the power source. When the die is rolled on a metal surface the input and ground will connect and pull down the input. This is registered by the controller that awakens the transmitter from sleep mode and sends a code to a receiver.

4.7.2 Augmented Wheel

The wheel (as seen in Wheel of Fortune) is an alternative to dice, but is not used as often. It is easier to implement the augmented wheel than the augmented die because the position is fixed for a wheel. There are mainly two approaches to augmenting a wheel either the position is detected or when the area of the wheel is changed.

- The position could be mechanically decoded. A different binary pattern for every area is etched on the backside of the wheel and read. This is called slip rings.
- The areas can be color coded and read with one or more photodiode or phototransistor. Concentric rings can be subdivided according to orders of binary numbers, so any radius can be read as a unique binary number representing is rotational position.
- A rotational position sensor can be used to know the position of the wheel. They are very accurate but expensive.
- Accelerometers can be connected to the wheel and sense the angle and hence the position of the wheel.
- Rotational potentiometers can be connected to the wheel hub. They are usually limited to how much a potentiometer can be turned but they can be modified for multi-turn.
- An input can be triggered when the areas of the wheel change by a mechanical construction. A button can be pressed every time the wheel moves to another area.
- Encoders that are found in computer mice can be connected to the wheel hub to detect rotation. A wheel has slits in it and a light emitter (LED) and a light detector (phototransistor) is placed on both sides.

4.8 Select OPTION from SET OF OPTIONS

A player selects things for different reasons, probability, statistics, tactics and strategy being some of them. The task can be enhanced by presenting information about probability and statistics for the player as a support for decisions or by implementing Artificial Intelligence that is capable of making decisions for the player. This is rather complex and requires precise input to know what information to show.

A PDA with a copy of the board and several alternatives on what information to show is an alternative. Another is to have special tokens that are placed on the board to show information about that position.

4.9 Order PLAYER to perform TASK

Ordering a player to perform a task can be simply to indicate to a player that it is his/her turn to do a task but also to describe the nature of the task. It does not however stipulate how the task should be performed, e.g. what option to choose in a “Select OPTION from a SET of OPTIONS” task. There are a lot of ways to enhance this task which depend on how players are presented with information. This task is typically generated by the completion of another task (the movement of a piece in Chess in the case it is the other player's turn to move) but can also be triggered by players being uncertain that other players are aware of the current game state. However it is also necessary in games which do not have a static task cycle.

The output can be handled like the output from the FIND task. In addition an audio output device can be used to inform about tasks that has to be done. The STARS system uses loudspeakers and headphones to play either prerecorded messages or computer generated speech [Magerkurth04].

5 DISCUSSION & CONCLUSION

The two primary results reported in this paper are 1) the concepts of links that describe how tasks can be supported or automated by technology and 2) the method of identifying functional requirements of a game design.

As new technology is constantly introduced, the examples given in this document have a time-limited usability. However, the technologies themselves will probably continue to be available for the foreseeable future and the technology survey and the links between tasks and technology can thereby have a design value when wishing to have an overview of possible technological solutions.

The technique of performing task analysis of games to identify atomic tasks can be used to gather functional requirements. This procedure allows game design to first focus on design of gameplay and at a later stage consider technology. As technology naturally limits game design this has to be an iterative design process but the task analysis described in this report allows the gameplay design to be the initial consideration while supporting the optimal use of the chosen technology.

6 ACKNOWLEDGEMENTS

The results presented in this paper was produced as part of the IPerG intergrated project which is funded by the European Union through the IST programme (FP6, contract 004457).

REFERENCES

[Andersson03] Andersson, T., Christensson, J., Göransson, J., Orrenäs, C., Strömqvist, S. and Tornéus, C-G. (2003) Interactive Tapestry student project, http://www.cs.chalmers.se/idc/ituniv/student/2003/ubicomp/interactive_tapestry/Interactive_Tapestry.pdf (accessed February 24, 2005)

[Andersen04] Andersen, T., Kristensen, S., Nielsen, B. and Grønbæk, K (2004) Designing an augmented reality board game with children: the BattleBoard 3D experience (Interaction Design And Children; Proceeding of the 2004 conference on Interaction design and children: building a community; Maryland; Pages: 137 – 138)

[Andersson04] Andersson, R., Hagström, D., Samuelsson, J., Sandlund, A. and Sjödin, M. (2004) RÄSER student project, http://www.cs.chalmers.se/idc/ituniv/kurser/04/uc/projektforslag/4_Projektbeskrivning_grupp4.pdf (accessed February 24, 2005)

[Bergsten03] Bergsten, J., Dafgård, J., Kronhamn, F., Hammargren, P. & Thurnherr, S. (2003) iBrush student project. http://www.cs.chalmers.se/idc/ituniv/student/2003/ubicomp/ibrush/index.htm (accessed February 24, 2005)

[Björk01] Björk, S., Falk, J., Hansson, R. and Ljungstrand, P. (2001) Pirates! - Using the Physical World as a Game Board. Paper at Interact 2001, IFIP TC.13 Conference on Human-Computer Interaction, July 9-13, Tokyo, Japan.

[Callert03] Callert, H., Harmanen, T., Persson, S. & Torbentsson, S. (2003) Elementary student project, http://www.cs.chalmers.se/idc/ituniv/student/2003/ubicomp/elementary/Dokument/rapport_elementary.pdf (accessed February 24, 2005)

[Eriksson04] Eriksson, D., Björk, S. & Peitz, J. (2004) Internal Deliverable ID9.1: Survey of Viable Technology for Augmented Board Games, Internal Deliverable for IPerG, http://www.pervasive-gaming.org/ (accessed February 24, 2005)

[Haglund03] Haglund, M., Johansson, F., Kasum, M. & Mälarberg, K. (2003) GymAssistWear student project, http://www.cs.chalmers.se/idc/ituniv/student/2003/ubicomp/gymassistwear/ (accessed February 24, 2005)

[Invisibletrain] The Invisible Train.http://studierstube.org/invisible_train/ (accessed February 24, 2005)

[Ishii99] Ishii, H., Wisneski, C., Orbanes, J., Chun, B. and Paradiso, J. (1999) Ping Pong Plus, design of an athletic-tangible interface for computer-supported cooperative play (Conference on Human Factors in Computing Systems; Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit; Pittsburgh, Pennsylvania, United States; Pages: 394 – 401)

[IPerG] IPerG – Integrated Project of Pervasive Games. 2004. http://www.pervasive-gaming.org/ (accessed February 24, 2005).

[Kent01], Kent, Steven L., The Ultimate History of Video Games, Prima Publishing, 2001.

[Lundgren02] Lundgren, S. (MSc thesis) (2002) Joining Bits and Pieces - How to make Entirely New Board Games using Embedded Computer Technology. HCI/Interaction Design, IT University of Göteborg.

[Magerkurth04] Magerkurth, C., Memisoglu, M., Engelelke, T. and Streitz, N. (2004) Towards the next generation of tabletop gaming experiences (ACM International Conference Proceeding Series; Proceedings of the 2004 conference on Graphics interface; London, Ontario, Canada; Pages: 73 – 80)

[Mandryk02] Mandryk, R. and Maranan, D. (2002) False prophets: exploring hybrid board/video games (Conference on Human Factors in Computing Systems; CHI '02 extended abstracts on Human factors in computing systems; Minneapolis, Minnesota, USA; Pages: 640 – 641)

[Pillows] IT + TEXTILES projects. Interactive Pillows. http://www.tii.se/reform/projects/itextile/pillow.html (accessed February 25, 2005)

[Preece02] Preece, J., Rogers & Sharp (2002) Interaction Design. John Wiley & sons, inc.

[Salen03] Salen, K., & Zimmerman, E. (2003) Rules of Play, The MIT Press, ISBN: 0-262-24045-9

[Åresund03] Åresund, M., Eriksson, C. and Harup, A. (2003) CoverallSound student project.