C. Magerkurth, T. Engelke, M. Memisoglu: Augmenting the Virtual Domain with Physical and Social Elements. 1. International Conference on Advancements in Computer Entertainment Technology (ACM ACE 2004). Singapore, June 3-5, 2004. pp. 163-172, ACM Press

Augmenting the Virtual Domain with Physical and Social Elements

Towards a Paradigm Shift in Computer Entertainment Technology

Carsten Magerkurth
Fraunhofer IPSI
AMBIENTE
Dolivostraße 15
64293 Darmstadt, Germany
magerkurth@ipsi.fhg.de

Timo Engelke
Fraunhofer IPSI
AMBIENTE
Dolivostraße 15
64293 Darmstadt, Germany
engelke@ipsi.fhg.de

Maral Memisoglu
Fraunhofer IPSI
AMBIENTE
Dolivostraße 15
64293 Darmstadt, Germany
memisog@ipsi.fhg.de

ABSTRACT

In this paper, means of enriching computer entertainment experiences by emphasizing physical and social game elements are discussed. A conceptual framework in which the relations between the virtual, the physical, and the social domains are modelled is presented. Interfaces that mediate between the domains are discussed along with a complementary software architecture that helps developing hybrid computer games. Finally, sample games that follow the approach of physical and social augmentation are presented.

Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentations**]: Multimedia Information Systems

General Terms

tangible interfaces, hybrid environments, tabletop games, computer games, entertainment

1. INTRODUCTION

Since the beginning of the home computer era in the late seventies, the term computer entertainment has mostly been used synonymously with computer or console games. For the first time, these games have allowed the players to immerse into a virtual game world, where the effects of game actions were only limited by the imagination of the programmers. When computers became more powerful, rich audio and visual presentations as well as more complex simulations enhanced the gaming experiences and turned the computer games industry into a mass market. Today, computer games have even become an innovating force that causes rapid advancements in other technological areas such as computer graphics or network technologies. While technology is advancing rapidly, the advancements and innovations in game concepts and contents seem to fall behind. There is an unsatisfactory trend of convergence among today's computer and console games, where most titles fall in narrow, well-defined categories such as first person shooters or real time strategy games. This might indicate that the design possibilities within the traditional computer entertainment paradigm have now been well explored and only few true innovations are still to be expected.

1.1 Beyond the Virtual Domain

We believe that by augmenting the mostly virtual nature of traditional computer games, new and exciting forms of computer entertainment can emerge.

1.1.1 The Social Experience

A first and obvious addition to the traditional paradigm is the notion of the social experience. Many other forms of entertainment (e.g. sports or board games) heavily rely on human factors to create a joyful interaction experience. In fact, most of today's computer entertainment titles are multiplayer and draw their attraction from human-to-human competition and cooperation. However, contemporary entertainment technology does not commonly incorporate social interaction as an integral part of the entertainment experience.

Regardless of the number of players involved, playing computer games is still being perceived as a mostly isolated activity [15]. This is due to the interaction being technology centered. Players do not interact with each other in a natural fashion, but their notion of each other is communicated through a computer display. Likewise, the actions they perform on each other are merely conveyed by joypads or keyboards. The richness of human-to-human interaction involving eye contact, mimics, and gestures is far from being captured in the purely virtual gameplay of computer games.

Co-located playing, as many consoles offer or is exerted at LAN parties, is also hampered by players sitting beside each other and interacting primarily with screen and joypad or keyboard. At least the communications overhead of remote play is reduced by such co-located multiplayer gaming, since verbal interactions become feasible[1] instead of chats or other

[1] Provided that the number of players is appropriate.

Figure 1: Co-located play with human centered (left) and technology centered (right) interaction.

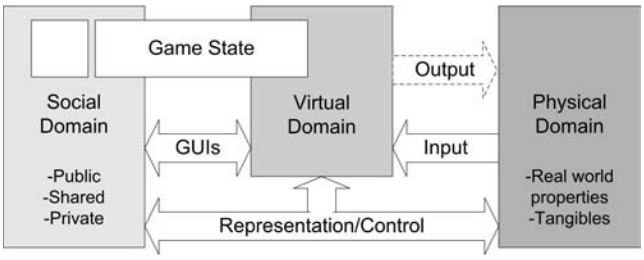

Figure 2: A conceptual model of augmented gaming applications.

indirect channels. Nevertheless, the social richness of co-located computer gaming as it is today is far from being near the richness of other popular game types such as board games. This is illustrated in figure 1, where the board game session (left) creates a much stronger social situation than the computer game session (right).

To further augment the social experience of computer games and facilitate a strong emotional involvement among the players, direct face-to-face interaction should be enabled and new interfaces between the players and the virtual domain must be introduced.

1.1.2 *Physical Interfaces*

These new interfaces are to ensure both the group situation to remain socially adequate and the transition from and to the virtual domain to be performed effectively. Human attention should remain focused on the social situation and not shift towards the interface to the virtual domain. We thus propose interaction metaphors that adhere to well-proven physical interfaces found in strongly human centered entertainment genres such as tabletop or board games. These physical interfaces should mediate the interaction between the human players and the virtual representation of the game world.

For instance, placing and orienting playing pieces that represent a group of dungeon explorers directly on a game table surface is more natural than doing the same with a traditional graphical user interface (GUI) in a computer game. This is especially relevant, when players are supposed to perform together and need to synchronize their actions. By jointly modifying the state of the game like tabletop gamers do, the social group dynamics of board games can be preserved and applied to computer entertainment. At the same time, the copious possibilities of computer interfaces can be complementarily integrated, for instance via large public displays or small private peepholes into the virtual dimension, provided that they do not demand exclusive attention and thus destroy the social situation. Together with the highly dynamic nature of computer simulations, the sensible use of the audio and visual presentation capabilities of entertainment technology has the potential to provide much more immersive and richer gaming situations than traditional tabletop media ever could.

We therefore propose to augment traditional entertainment technology with social and physical elements to form a new class of hybrid gaming applications. These hybrid applications should integrate the social dynamics of co-located groups with computer games via interfaces that do not distract from the group situation. In the following section, we will outline our conceptual model that explains how hybrid gaming applications can integrate game elements from virtual, social, and physical dimensions.

2. A CONCEPTUAL MODEL

In contrast to traditional computer entertainment, augmented gaming applications define game elements from the physical and the social domains as integral parts of the gaming experience. The relationships between these domains is illustrated in figure 2.

2.1 The Virtual Domain

The predominant domain in traditional computer games is virtual, i.e. the entire game takes place inside a micro-controlled system. A digital game logic sets up and maintains a virtual game world that is perceived and controlled through a GUI. Each player uses a GUI to access a segment of the game state that is relevant for him in the given game context. In an augmented game, segments of the game state can also be outside the computer and a GUI becomes only one of several means to modify and retrieve game information within the virtual domain.

2.2 The Social Domain

Face-to-face group settings with natural means of interaction between players inevitably create social situations. By the use of directed speech, gestures, and mimics, a social domain emerges in which parts of the game state can reside that are only partially (or not at all) reflected in the virtual segments of the game state. For instance, in games with a strong emphasis on cooperation *and* competition, diplomacy and alliances will be key elements of the gaming experience, but they might entirely be of social nature without the virtual domain necessarily being aware of them. Virtual interfaces such as private GUIs might be used for clandestine negotiations, but apart from the communications functionality, the virtual domain would still be unaware of the players' state of the diplomacy. Nevertheless, game state information could also be shared among both domains, e.g. a social alliance could result in virtually forging a pact to allow controlling the partner's troops or sharing virtual resources.

A related aspect of the social domain is the notion of private, shared, and public information. When cooperation is important in a game, individual information will be made public in order to maximize the group's efficacy. Complementarily, competitive gaming situations cause individual information to be kept private, so that no competitive advantages may be lost. In intermediate situations, some information might be shared to only some of the other players and held back or faked towards others. While the virtual domain might not necessarily be aware of players sharing information between each other, the distinction between public and private interfaces across both domains is essential to enable public, private, and shared information in the social domain. Therefore, both private and public GUIs should be available to foster social dynamics. Whenever game information is transferred from a virtual segment of the game state via a private interface, it is up to the player to decide whether to make public, share, fake, or hide the information in the social domain. Likewise, it is up to the social skills of the other players to believe, to doubt, or to mistrust the given information and act accordingly.

2.3 The Physical Domain

The physical domain consists of the world around us. We mainly see two aspects of the physical domain to be relevant for gaming applications. First, there are real world properties such as illumination, noise level, the players' positions, or even time (although the latter is also represented in the virtual domain). Second, there are special physical or tangible interfaces that effectively mediate between the virtual and the social domain.

2.3.1 Real World Properties

The integration of real world properties can add to the immersiveness of a gaming experience, especially when atmospheric properties such as background audio or the illumination level are utilized. Think of a vampire game where the light conditions at the game table directly relate to the danger of being attacked by the undead. Or a spy game, where real world noise makes it difficult to overhear a conversation between non-player-characters in the virtual domain.

When it comes to letting the virtual game logic consider real world properties, there are numerous ways of integrating different kinds of modalities via dedicated sensing technology, e.g. [1]. The challenges here are clearly more related to making optimal use of the real world data than gathering it and transferring it to the digital game logic.

The way back from the virtual to the physical domain is much harder to take than vice versa, hence the dotted line in figure 2. Naturally, a computer has only limited means of changing the real world around us. However, creating atmospheric audio or changing the illumination level through an electronic dimmer are only two feasible examples. By utilizing network controlled hardware interfaces such as power sockets, many real world context generating devices can be regulated. Just imagine *really* smelling the gore of a slain zombie.

2.3.2 Tangible Interfaces

As outlined in the introduction, for a truly social gaming experience to emerge, we need interfaces to the virtual domain that do not bend the players back to technology centered interaction styles. We opt for tangible interfaces such as physical game boards that fuse state representations with means of state manipulations. As [14] outline, the principal attribute of a tangible interface is the seamless integration of *representations* and *controls*. That is, an ideal tangible interface does not differentiate between input and output as a GUI does, because by manipulating a physical state, this state already *is* its own representation. Tangible interfaces are thus much more direct and natural to use than a GUI, whenever a possible virtual representation matches the physical representation of the interface. For instance, opening a chest in an augmented computer game can simply be achieved by opening a physical model of the chest on a table surface. Moving and turning the chest corresponds to moving and turning the physical control. Any physical action that is imposed on the control also updates its representation, i.e. the chest obviously is and remains open or turned.

A typical tangible interface is of public nature, i.e. each player should normally be able to perceive and alter the physical state of an interface component such as the position of a playing piece. Therefore, the tangible interfaces' representations are equally relevant for both the social and the virtual domain. The transfer to the social domain is achieved by the group members simply looking at (or touching) the interface, whereas the virtual domain has to utilize appropriate sensors to become aware of the physical states. Similar to purely social game elements that have no virtual representation, not all physical states have to be transferred to the virtual domain, either. The shape or color of a playing piece might, for instance, have a meaning for the group, whereas the computer only understands its position and identity. Likewise, a representation might in fact be more complex in the virtual than in the physical domain. For example, the virtual game entity belonging to a physical playing piece might have a multitude of virtual attributes such as strength, intelligence, or hit points that are conveyed to the social domain via GUIs and not through the physical representations.

Please note that in figure 2 the game state is not represented in the physical domain. This is because each game relevant property of the physical domain is either transferred to and thus represented in the social domain (players see the color of a playing piece) or in the virtual domain (the computer senses the position of a playing piece). Due to the mediating nature of the physical domain, it thus consists entirely of game state information that is either duplicated/ redundant or irrelevant. Therefore, no unique segments of the game state can exist in it.

2.4 Implications of the Model

In this section we will outline how to apply the model to the design of augmented computer games. We will discuss which kinds of games are suitable for a hybrid platform setup and how to design game elements that take advantage of the capabilities offered by a hybrid platform.

2.4.1 Scope

A fundamental issue regarding the applicability of the proposed model is the scope of game types that can be realized. First, games need to be suitable for multiplayer use. The more predominant the multiplayer portion is, the easier it will be to create a strong social situation. Second, games should not entirely rely on real-time GUIs that demand unshared, permanent attention such as e.g. in space combat simulators or jump n'run games. However, neither GUIs per se nor real-time gameplay are problematic, whereas only their combination hinders human centered interaction forms and thus works against the social domain.

Naturally, games that deal mostly with information that is hard to convey via physical interfaces will make more use of GUIs than games that borrow elements from the physical world, such as *Sim City*, *Panzer General*, or puzzle games. Whenever physical and virtual properties of the same game element complement each other, GUIs and physical interfaces will make up for an ideal combination.

While turn-based games are proven to be suitable for social groups (think board or tabletop games), there is no reason not to make use of real-time gameplay. Due to the time-multiplexing problem of mouse driven UIs [10], well designed tangible interfaces are in fact much faster to use than many plain GUIs. Care must be taken, though, to keep the pace of the game manageable for group interaction.

2.4.2 Design Issues

Developing games for hybrid setups requires certain considerations that go beyond traditional computer game design. First and foremost, there are several domains to take into account, and care must be taken to distribute game elements along these domains. Some game elements clearly belong into one domain or another, but in many cases it is a matter of deliberate choice.

For instance, in traditional computer games, random number generation for achieving a variable game flow is typically and easily performed by the computer. However, when we add physical and social game domains, it might be sensible in certain game situations to adopt techniques from the real world, e.g. creating random numbers via the physical rolling of dice. For most players, the act of rolling dice is a highly social activity involving skillful rolling techniques which are permanently supervised by the other players to prevent cheating. It all depends on the social dynamics of a game situation whether "too lucky" dice rolling is tolerated, appreciated, or condemned. We thus have to weigh the advantage of unobtrusively creating randomness in the virtual domain against potentially exciting social effects when the physical dice interface is seemingly manipulated by one of the players.

Another important area of hefting domains is artificial versus human intelligence (AI), including dialogue and reactive behavior. While the advancements in the field of AI make actions of virtual entities more and more believable, they still pale against the richness and immersiveness of the social interaction with a skilled human game master or storyteller. On the other hand, putting too much action into the social domain might incumber the creation of interfaces between both domains and lead to gaming styles that are essentially identical to e.g. traditional tabletop role playing games that work very well without any computer support.

Finally, for many simpler games it might be beneficial not to implement the complete game rules into the virtual domain, even though the computer is perfectly suited to e.g. judge a *Backgammon* session. If we simply provide a basic virtual and physical platform that models the relevant game objects, but only few relations between them, many game rules and their enforcements can be left to the social protocols of the players around the table. Flexibly re-evaluating rules in the context of a particular game situation might be enjoyable and the development of house rules is far easier, if no computer logic denies certain rule variations. On the other hand, transferring game rules from the virtual to the social domain requires players to be sufficiently familiar with them, which might or might not be the case.

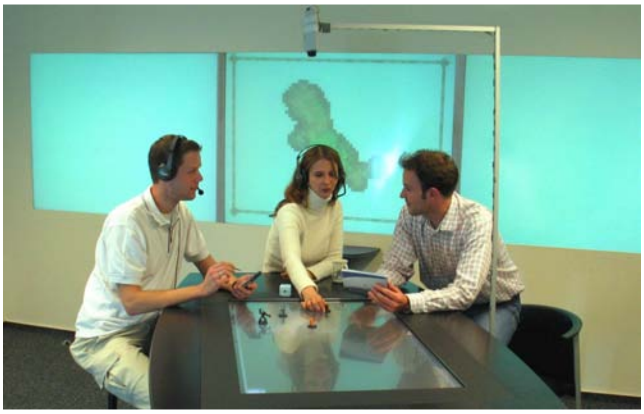

Figure 3: Adapted GUI devices in a Roomware environment

3. APPROPRIATE INTERFACES

The key to realizing augmented gaming applications lies in the development of new, appropriate interfaces. Following our model, we distinguish between interfaces that involve the virtual and the social domain (GUIs) as well as interfaces that are focused on the physical domain. Both types of interfaces can complement each other and either one may be more suitable depending on the current context of the game application.

3.1 Adapted GUIs

The GUIs that we think to be especially suitable for colocated group games are standard GUIs from a technical point of view, but we try to use them in a human centered interaction style. We argue for unobtrusive and natural GUIs as we have described in our works about so called *Roomware* [13]. Following the Disappearing Computer paradigm, such Roomware interweaves computer technology with room elements like e.g. tables or walls. Roomware components typically embed touch sensitive displays that are operated with pens or plain fingers.

Using several Roomware components, we have created an experimental setup for gaming applications [5] called *STARS* that integrates specialized GUI Devices for dedicated gaming purposes. For instance, the game table shown in figures 3, 7, and 9 is the InteracTable Roomware component. Its touch screen is used for displaying game content that can naturally be manipulated by the players' fingertips. Also, a large vertical display (the DynaWall) can be utilized to show public information that each player should be able to view at any time. This could include a permanent overview of the individual players' scores to instigate rivalry or the advancements that the entire group has made, if the game content is of a more cooperative nature.

To create private, shared, and public information in the social domain, it is essential to provide additional private interfaces to the virtual domain. Personal digital assistants (PDAs) are thus important GUI devices that are used to administer private information or enter private commands. Consequently, the PDA's most important function is to serve as a generic, private input and output device as well as a means for private communication between players (both being shown in the video footage). Any player may send short

Figure 4: A custom game board sensing magnetic fields (left) and a touch display with camera recognition (right).

messages to other players via the PDA. This provides the simple advantage over other communication channels in a co-located setting that even if the act of writing a message might or might not be hidden from other players, the message's addressee, and not just the message itself, becomes covered.[2] The Viewport, a special PDA that also serves as a physical interface, is described below.

3.2 Physical Interfaces

Physical interfaces can mediate between the social and the virtual domain. In our approach, we focus on game board interfaces that detect several properties of tangible game objects. By using game boards and tangible game objects, the social richness of traditional tabletop games can be preserved, because the state of the interface is reflected directly within the virtual *and* the social domain.

While the direct transfer between the physical and the social domain is trivial, the virtual domain needs to utilize physical sensors to become aware of the state of the tangible interface. We are exploring several technologies that can be used to sense properties of game objects.

The most obvious property is the game object's position which can be discrete or continuous. Other interesting properties include the object's identity, orientation, or state (such as open vs. closed). No single sensor technology can deal with each of these properties equally well. It therefore depends on the requirements of a concrete gaming application to prioritize among the pool of properties and select the most suitable technology accordingly.

3.2.1 Camera Recognition

For the STARS setup described above, we have positioned a camera over the game table (see figure 3). Physical objects placed on the table surface are detected by the camera in addition to manipulating virtual information on the touch display with fingers (see figure 7). The corresponding image analyzing software (shown in the video footage) is capable of identifying an arbitrary number of objects at discrete positions. Due to varying light conditions it is, however, not easily possible to identify objects based on their color. The resolution of typical webcams (640x480 in our case) also inhibits the detection of object orientations, unless their shapes are designed for being visually analyzed. Even worse, suitable cameras are mostly expensive and setting them up is a shaky and time consuming effort.

[2]Clandestine communication is an irrelevant issue for remote network games and is thus one of the few advantages of isolated gaming.

Even though camera recognition obviously involves serious drawbacks, there are also unique advantages such as the easy combination with graphical displays, so that dynamic and visually rich game boards can be realized with game objects placed on the display (see figure 4 right) effectively blending GUIs with tangible interfaces. Another advantage is the generic nature of visual recognition. Any kind of object can be detected, so that arbitrary playing pieces that are not tagged or otherwise manipulated work well. Also, the results of dice rolling can easily be transferred to the virtual domain, if the dice are sufficiently large (see figure 4 right). And for the STARS setup, we even take the physical positions of the players into account. When a player reaches over the table, a simple algorithm determines the side of the table the camera image was first disturbed and allows the software architecture's *Presentation Manager* (described in a later section) to adapt the display accordingly.

3.2.2 RFID Transponders

A very robust albeit slightly expensive enabling technology is radio frequency identification (RFID). Game objects can be tagged with small (cheap) RFID transponders and detected by appropriate (expensive) antennas. When these antennas are placed into a game board, discrete positions and object identities can be detected. Furthermore, most transponders can also store a few bytes of state information. We use RFID for both the Candyland adventure platform (see figure 8) and for our Viewport devices (see figure 5). Additionally, the InteracTable also contains an RFID antenna that detects physical tokens placed on the table surface. We currently use these to initiate and terminate game sessions in a natural way by simply placing a tag on the table to begin, and removing it to end and save a game session. This way, the players avoid dealing with mouse or keyboard to get a game going.

3.2.3 Magnetic Fields

Magnetic game objects can be sensed very inexpensively by magnetic field sensors built into a game board (the left part of figure 4 shows our own magnetic sensor board). While the position recognition works robustly, magnetic field sensors can not be used to determine a game object's identification or orientation, unless more expensive and tricky to use hall effect sensors are applied. Nevertheless, if economic factors play a role, magnetic field sensors might be the first choice. In fact, practically all electric chess boards commercially available make use of magnetic field sensors.

To tackle the missing object identification, the virtual domain must rely on certain assumptions constraining the use of the game board. For instance, no two objects must be removed from the board at the same time. Also, the virtual domain must have some means of conveying to the players where a certain object should be placed. When there is no GUI available, light emitting diodes (LEDs) can form a cheap and effective channel of output from the virtual to the physical domain (see figure 4, left).

3.2.4 Other Physical Interfaces

There are numerous other means of linking physical and virtual properties which we have not yet explored. Other technologies include light sensors, audio sensors, force sensors, or proximity/ vicinity information through passive IR. A popular multi-sensor and communications platform is [1].

Figure 5: Viewport devices used with tagged playing pieces.

3.3 Viewports: Hybrid Interfaces

Viewport devices are special private interfaces that integrate both a physical interface and a GUI in one device (see figure 5 or the video footage). The Viewport consists of a PDA equipped both with 802.11b and an RFID antenna that detects physical objects such as playing pieces tagged with appropriate transponders. It comes from the EU-IST project Ambient Agoras [12]. The Viewport is geared towards natural interaction styles; operations on physical artifacts are easily performed by pointing at them.

Figure 5 (left) shows a Viewport displaying information about a tagged object that belongs to the owner of the Viewport. Figure 5 (right) shows the same Viewport displaying information about an object belonging to a different player. Obviously, the information transmitted from the virtual domain consists of a varying degree of detail (concrete stats versus a cursory description). In this context, the interesting part of the Viewport device lies in the combination of its *public* physical and *private* virtual interface. While the social domain gets to know that objects are investigated or manipulated, the concrete action performed is not made public. This allows for highly interesting social situations as alluded to in figure 6 and mainly in the video footage: By pointing at a physical model with a Viewport, a player searches a statue for a hidden treasure. Due to his high skills, he indeed finds a diamond ring he wants to keep and hide from the other players. However, the public nature of the Viewport's physical interface causes awareness about the preceding action among the others, and thus evokes the necessity to explain them the action he just performed.

4. SOFTWARE FRAMEWORK

To facilitate the creation of augmented computer games, we have developed a software framework called *CATS*. The CATS framework integrates the different kinds of interfaces described above and translates our conceptual model to a re-usable code base. It was used for the creation of our first generation of demo games described in the next section.

The goal of the CATS framework is to reduce the added complexity introduced by the physical and social dimensions. We use several modular code components (*"Managers"*) that each tackle a specific game relevant area at a rather abstract level. The CATS components build upon lower level engines that are dedicated to typical tasks also relevant for traditional gaming applications, e.g. network, 3D/ 2D graphics, audio, etc. For the lower level engines we use proven third-party middleware whenever possible (e.g. GapiDraw for mobile and stationary 2D graphics or OpenAL for audio). The architecture is almost entirely based on the .NET platform to allow a flexible distribution of components among different computers.

Figure 6: Information about a game object conveyed through a private interface.

We will now describe those components that are especially relevant for the notion of physical and social augmentation. Other components include more mundane functionality for managing users or providing a generic session management. They are omitted here.

4.1 Framework Components

4.1.1 Device Manager

The Device Manager is the central component that administers the supported hardware devices. It maintains a set of device and service descriptions for each supported device class. Furthermore, it keeps track of the set of devices currently integrated into a game session. It fundamentally distinguishes between *GUI Devices* and *Physical Interface Devices*.

GUI Devices such as the Roomware components described above are computers equipped with displays and corresponding input devices, typically they are PCs or PDAs. For all GUI Devices, certain basic assumptions apply, i.e. they are able to render graphical output, are suitable for GUI dialogs (joint input and output), and their state can be completely controlled by an application. Relevant device descriptions for GUI Devices include technical parameters such as display resolution, their presumed orientation (e.g. horizontal or vertical), and most importantly, their private or public nature. PDAs, for instance, are typically private. Another component, the *User Manager* keeps track of the active players and associates private devices with corresponding players.

Physical Interface Devices such as the various game boards from the preceding section exchange a dedicated set of parameters between the physical and the social domain. There are no assumptions to be made about them, especially their state can not necessarily be constrained by a software application. They are characterized by their dedication and details about their parameters.

For both GUI Devices and Physical Interface Devices there is a set of *Interface Services* available that can be queried by an application via the Device Manager. Interface Services are implemented for every supported device using Remote Procedure Calls. Interface Services are the workhorses regarding the actual creation and presentation of interfaces. The functionality of Interface Services available for each device varies with the characteristics of the device. Output and dialog services currently rank from simple DISPLAY() or MESSAGE() to several more complex MENU() variations. They are mostly available (or exclusively such as WINDOW()) for GUI Devices. What a service such as DISPLAY() actually does, depends on the specific device characteristics. Input services include the gathering of physical properties as well as keyboard and mouse related functions for GUI Devices. Those Interface Services that exist for both GUI Devices and Physical Interface Devices work equally and independently of the actual device. For instance, one service, TILE(), can update the graphical representation of an area on a game board. With a physical device such as the board on figure 4 (left) it results in the blinking of LEDs, whereas the GUI on figure 4 (right) is updated with new bitmaps.

The most beneficial effect of maintaining device characteristics and service descriptions in the Device Manager lies in the simplified programming model for the game application programmer. Instead of having to deal with a concrete setup of devices, the developer can simply query for, e.g., a private GUI Device to send a player a clandestine message. As long as the required Interface Services are implemented for a given device setup, the application can automatically adapt to changing device configurations.

4.1.2 Game State Representation

The virtual domain maintains the virtual segments of the game state (there may be additional social segments, but, by definition, the virtual domain is unaware of them). Depending on the current setup of devices as maintained by the Device Manager, certain game properties such as object positions may be integrated from the physical domain. In another setup, they may again be virtual or imported by other physical interfaces. To facilitate the virtual representation of the game state for the programmer, we have introduced an abstract data type called ChyProperty (hy for hybrid), from which several derived classes for different types of data exist. ChyProperties may either contain normal data programmatically written to them or, if available, subscribe to the output of physical interfaces mediated by so called *Mediators* that translate the potentially unreliable, raw data to the constraints and dimensions of the ChyProperty. By doing so, hybrid game entities can easily be built that contain both physical and virtual properties without the application developer having to care about the conceptually different types of properties. In effect, it becomes irrelevant for an augmented game whether, for instance, it utilizes a physical game board or an ordinary GUI as an interface.

The second function of ChyProperties is related to fostering different degrees of privacy in the social domain by communicating via public and private interaction channels from the virtual domain. Each ChyProperty can be tagged with a Privacy Index that is a measure for the corresponding level of privacy. This makes it especially simple to realize game situations as shown in figure 5 and 6, where certain properties of an entity are queried by a private interface. The information shown in figure 5 (right) might have been more lavish, if the players perception roll against the Privacy Indices of other, more private properties had been more successful.

4.1.3 Presentation Manager

To support rich co-located interaction between the players and the game media, a horizontal layout of the game media with the players sitting around it is most favorable, because such a setup allows easy interaction with both the human players and the game media. If the Device Manager identifies the central media as a horizontal, public display, such as the InteracTable, the game application may choose to invoke the *Presentation Manager* to provide a GUI adapted for the group setup.

The positions and viewing angles of the players have always been problematic in tabletop games, i.e. someone sitting at the left side of the table perceives a very different image than someone at the opposite right side. Previous solutions to this orientation problem were based on abstract, un-oriented game boards and playing pieces that look mostly the same from any viewing angle [6]. The CATS Presentation Manager however, can skip un-oriented game objects and rotate anything displayed on the GUI around its own axis to that angle the current player can perceive best (for a demonstration, please see the video footage). This includes window objects that additionally grow in size with rising distance to the current player ensuring optimal visibility. Also, graphical objects belonging to other players can automatically become half-transparent to prevent obscuring parts of the board.

To use the interface orientation functionality, a game application can hint about whether single graphical objects should be sensitive to orientation (and if the orientation should be limited to steps of 45, 90 or free), whereas performing the orientation itself is completely performed by the Presentation Manager which is invoked by the respective Interface Services. Whether the auto-orientation works well for a given game largely depends on the amount of detail of the game objects. When large amounts of text are displayed, changing the orientation is usually appreciated much by the players. For more primitive game objects, the benefit of rotating objects is less significant.

5. EXPERIENCES

5.1 Realized Games

We are implementing several game prototypes that conceptually follow our ideas of enriching game experiences by strengthening the role of additional dimensions. In this section, we will describe three examples that are based on our framework.

5.1.1 KnightMage

The augmented game KnightMage (see figure 6 and 7) implements a basic set of rules for medieval hack'n slash style role-playing adventures. The players explore and ransack dungeons and landscapes filled with horrifying monsters, moderated by a human Game Master (GM). While the game progresses, the players solve riddles and find treasures or weapons to help them against the countless enemies. The idea behind KnightMage is to combine the strong social sit-

Figure 7: A touch sensitive table surface displaying the KnightMage game board.

uations of traditional tabletop role-playing games with typical benefits found in computer adaptations of role-playing games (which, however, typically lack the immersiveness of the pen-and-paper versions). What makes KnightMage especially interesting is the mixture of cooperative behavior when fighting monsters and the competitive interests when searching for treasures that each involve very different interaction modes. In KnightMage, certain game events are privately communicated from the game logic to a player, e.g. a players character might hear or find something and decide on her own, if she lets the others know or not. This is shown in figure 6, where a private GUI informs a player about a hidden treasure that she just found.

Another focus of KnightMage lies in the flexibility of interface integration. Both standard GUIs and physical interfaces are supported through the Device Manager, so that e.g. the game board can either be rendered on a standard GUI, on a hybrid interface such as an interactive table surface (see figure 7 or an entirely physical interface (see figure 6). This allows for researching the effects on the gaming experience of either stressing the benefits of virtual representations and GUIs (rich audio and video, highly dynamic and changing game boards etc) or stressing the benefits of physical interfaces (direct control and manipulation, no attention drain etc). A controlled study has not yet been conducted, but we will explore the effects of balancing interfaces in the future.

5.1.2 Candyland

Our second research prototype is a hybrid gaming application for children called Candyland (see figure 8). Candyland represents the model of a small village, where physical objects such as inhabitants or houses can be freely placed by the children. An RFID based sensor interface integrated in the game board synchronizes the positions of the physical objects with their virtual counterparts. The game logic in the virtual domain runs a simple adventure/ storytelling game engine that talks to the children (i.e. plays digitized sound samples) depending on their movements of the playing pieces.

Physical laws also apply to Candyland in that the speech of its inhabitants is spatially correctly conveyed through a

Figure 8: The Candyland adventure platform.

surround sound system depending on their positions. Both line-of-sight and radiation-of-sound algorithms to deal with blocking elements such as houses are available. The children are encouraged to experiment with placing buildings and inhabitants freely on the board and listen to the effects of their spatial manipulations.

The aim of Candyland is to observe how children make use of and appreciate a hybrid gaming application that combines digitized adventures and stories with interfaces they are familiar with (building bricks). As a business model, it might be interesting to offer the physical game board separately (perhaps at a reduced price) and then bring out several sets of adventures and stories as well as game mats and playing pieces (each with different themes and complexity). This would open two possible ways of creating revenues for the manufacturer, i.e. traditional entertainment software and physical collectibles which are collected for their own sake.

5.1.3 Monopoly Adaptation

As a sample application for the STARS platform, we have also developed an adaptation of the classic Monopoly board game from Parker/Hasbro. STARS M*nopoly (see figure 9) attempts to make optimal use of the capabilities that a Roomware environment offers. Because of the various Monopoly adaptations for personal computers, our STARS M*nopoly aims to become a testbed for comparing pure virtual vs. pure physical vs. mixed implementations of the same game.

In STARS M*nopoly, the mundane tasks of shuffling and stacking cards are discontinued, because random events are directly displayed on the virtual game board. Due to the virtual representation of game money, there are several statistical functions available. On the wall display, a permanently updated diagram is shown that gives an overview about the financial development of each player and thus facilitates a strategic planning of buying and selling properties as well as it instigates competition among the players. Due to the high amount of textual information on the game board, STARS M*nopoly profits greatly by the Presentation Manager's functionality to rotate information objects to the current player's viewing angle.

As an optional addition to the original rules, money can be clandestinely transferred between players, so that secret alliances against the leading player are fostered. This in-

Figure 9: A Monopoly Adaptation.

troduces an interesting new game element that stresses the proceedings in the social domain by forcing players to weigh competitive versus cooperative behaviors in an otherwise purely competitive game.

5.2 Preliminary Evaluation

We are currently investigating new game concepts and are constantly developing new games with the our programming framework. While our own experiences with the games we have created so far are very favorable and support our vision of integrating the best of social, virtual and physical game elements, a public test is necessary to gain real-world validity.

During a public open house day, we had the opportunity to observe and videotape eight different groups of 11-14 year-old girls interacting and playing with our setup of games over the course of a day. We chose not to test hypotheses in a controlled experimental setting, but conducted a formative evaluation in which we "let the girls loose" and observed their interactions during the games as well as asking them about their experiences and suggestions afterwards.

In general, most subjects were very excited and appreciated our setup of games. Also, as intended, their interaction mostly remained human centered with a lot of activities going on in the social domain. While some of the new game concepts we introduced were immediately understood by most, it took a while until the girls adopted them and e.g. made sensible use of the clandestine communication means. We attribute this to the rather short period of time each group had to get acquainted with the games. Both physical and graphical interfaces were understood immediately, since many girls already had experiences with computers. Most girls were impressed with the presentation richness of the virtual interfaces both including the visual quality and the audio output.

Regarding physical interfaces, we had underestimated the level of stability needed for some of the interfaces, especially for the camera recognition in KnightMage. The camera above the game table was fixed firmly enough for our own use, but some groups shook and bumped either camera or table so badly, that we had to re-calibrate the camera recognition software during a running game session. Beforehand, we had invested quite some effort to let the software cope with varying lightning conditions, but did neglect the image alignment issue. In addition, we had not anticipated the target group of our research being half as tall as we are. Due to the somewhat large dimensions of the game table hardware, some girls stood up to reach the edges of the game board, disturbing the camera recognition. For everyday use in a real world setting, more robust sensing technologies such as RFID sensors seem to be more favorable than visual recognition. Most suggestions the girls made were directly related to glitches and quirks with our sample games and have been integrated into the current versions.

We will address some of the more specific research questions with controlled experimental studies in the future. Two of the most interesting issues to deal with will be the direct comparison between virtual, physical, and mixed implementations of the same game as well as the effects of complementarily using virtual and physical interfaces for the same game.

6. RELATED WORK

Both academic researchers and professional game developers have worked towards augmenting gaming experiences with elements beyond traditional computer entertainment.

The first commercial foray in this direction dates back to 1987. Robot Rascals from Ozark Softscape was advertised as the "First Great Family Computer Game". It integrated a deck of physical playing cards with a unique mixture of turn-based action and strategy. A group of robots was set out to hunt rare artifacts on a distant planet with the map of the planet being inside the computer. Robot Rascals was designed as a true multiplayer game that offered no single player mode. Alas, the link between virtual and physical world was quite limited in Robot Rascals. The state of the "luck cards" had to be entered manually into the computer, which plays somewhat oddly. Robot Rascals failed to attract a large audience, even though the game definitely is very interesting to play.

The second and less well designed commercial approach is from the most famous makers of interactive fiction, Infocom: Fooblitzsky is a computer augmented board game in which each player is a dog moving on the game board to obtain a desired set of items. As a hybrid game, Fooblitzsky can not convince (dice rolling is purely virtual etc). The ugly monochrome graphics and the unbalanced gameplay do not stand the test of time.

The final commercial products we are aware of are Zowie Intertainment's smart toys that are described in [11]. These smart toy systems sensibly mingle computer games with physical toy representations. The toys are designed to integrate insights from child psychology to help children learn through play.

The academic research project False Prophets [6] is a hybrid board game, in which players jointly explore a landscape on a physical game board. A custom crafted infrared sensor interface helps identifying the playing pieces, while the game board is projected on the table. The realization of False Prophets is similar to parts of our platform, even though it is currently limited to a single exploration game. Recently, False Prophets switched their hardware platform to a *Diamond Touch* multi-user table that is also used by [8] in the research of collaborative games for multiple users on table surfaces.

Bjork et al. [2] presented a hybrid game system called Pirates! *(sic!)* that adds the world around us to gaming applications with players moving in the physical domain and experiencing location dependent mini-games on mobile computers. Thereby, Pirates! follows a very interesting approach to integrate virtual and physical components in game applications. Unfortunately, the mini-games on the PDAs do not involve multiple players, so that social aspects are not very relevant for Pirates!.

Another location based gaming application with a similar approach was presented in [9]. The focus of the Schminky game project is to employ a sound based game for handheld computers in a cafe in Bristol, UK, and gather real-life data from groups of caf guests. Apart from the very unique game play, Schminky is also interesting in terms of the exemplary design process. While Pirates! stresses more physical elements and less social ones, Schminky has an opposite focus.

A recently initiated research prototype is the Smart Jigsaw Puzzle Assistant [3]. It is an augmented physical jigsaw puzzle game with attached RFID tags to its pieces. RFID technology is used to solve the inherent needle-in-the-haystack problem by integrating a handheld RFID scanner that can track down the perfectly fitting piece in the pile of jigsaw pieces.

The Tangible Viewpoints system [7] is a beautiful tangible table interface for multimedia storytelling applications with a very rich presentation and tangible objects. However, its focus is limited to interactive storytelling and it does not explicitly support social aspects e.g. by providing means for private interaction.

Ishii et al [4] have created a hybrid gaming application called PingPongPlus that augments a traditional ping pong game table with several output modes such as sound and graphical effects projected at the table surface, making it a highly entertaining experience to watch and listen to. Apart from the rich presentation, PingPongPlus is a good example for linking real world actions to virtual representations.

7. CONCLUSIONS

We have presented our vision of creating future entertainment experiences by adding physical and social components to traditional computer games. We have discussed both a conceptual framework and a complementary software architecture that model the relationships between virtual, physical, and social dimensions. We have also presented our first realizations of physical interfaces and augmented games together with initial usage experiences.

In the future, we will continue to develop new interfaces as well as sample games to conduct controlled experimental studies that research the effects of hybrid games. Furthermore, it will be very interesting to augment our approach by taking physical activities of the players into account. This includes, for instance, live action roleplaying (LARP), where real world parameters such as players' positions and actions take a central role.

8. ACKNOWLEDGMENTS

We thank our colleagues and student co-workers Sascha Nau, Wolfgang Hinrich, Richard Stenzel, and Norbert A. Streitz for their valuable contributions and insightful comments on our work. This paper was supported by a grant from the Ladenburger Kolleg "Living in a Smart Environment" of the Gottlieb Daimler and Carl Benz foundation.

9. REFERENCES

[1] M. Beigl and H. Gellersen. Smart-its: An embedded platform for smart objects. In *Proceedings of Smart Objects Conference 2003*, 2001.

[2] S. Bjork, J. Falk, R. Hansson, and P. Ljungstrand. Pirates! using the physical world as a game board. In *Proceedings of Interact 2001*, 2001.

[3] J. Bohn. The smart jigsaw puzzle assistant: Using radio frequency technology for building augmented real-world games. In *Workshop on Gaming Applications in Pervasive Computing Environments at Pervasive 2004*, 2004.

[4] H. Ishii, C. Wisneski, J. Orbanes, B. Chun, and J. Paradiso. Pingpongplus: Design of an athletic-tangible interface for computer-supported cooperative play. In *Proceedings of CHI'99*, pages 394–401. ACM Press, 1999.

[5] C. Magerkurth and R. Stenzel. Computer-supported cooperative play - the future of the game table. In *Proceedings of M&C'03*, September 2003.

[6] R. Mandryk and K. Inkpen. False prophets: Exploring hybrid board/video games. In *Extended Proceedings of CHI 2002*, pages 640–641. ACM Press, 2002.

[7] A. Mazalek, G. Davenport, and H. Ishii. Tangible viewpoints: A physical approach to multimedia stories. In *Proceedings of Multimedia'02*. ACM Press, 2002.

[8] S. Pape, L. Dietz, and P. Tandler. Single display gaming: Examining collaborative games for multi-user tabletops. In *Workshop on Gaming Applications in Pervasive Computing Environments at Pervasive 2004*, 2004.

[9] J. Reid, R. Hull, T. Melamed, and D. Speakman. Schminky: The design of a caf based digital experience. *Personal Ubiquitous Computing*, (7):197–202, 2003.

[10] A. Schmidt and C. Decker. Physical and tangible user interfaces. In *Proceedings of M&C'03*, 2003.

[11] H. Shwe. Smarter play for smart toys. In *Zowie Intertainment White Paper No. 3208*, 1999.

[12] N. Streitz, C. Röcker, and T. Prante. Situated interaction with ambient information: Facilitating awareness and communication in ubiquitous work environments. In *Proceedings of HCI'03*, pages 133–137, 2003.

[13] N. Streitz, P. Tandler, C. Müller-Tomfelde, and S. Konomi. Roomware. towards the next generation of human-computer interaction based on an integrated design of real and virtual worlds. *J. A. Carroll (Ed.): Human-Computer Interaction in the New Millennium*, pages 553–578, 2001.

[14] B. Ullmer and H. Ishii. Emerging frameworks for tangible user interfaces. *IBM Systems Journal*, 39(3):915–931, 2000.

[15] J. Zagal, M. Nussbaum, and R. Rosas. A model to support the design of multiplayer games. *Presence: Teleoperators and Virtual Environments*, 9, September 2000.