The Magic Lens Box: Simplifying the Development of Mixed Reality Games

Richard Wetzel, Irma Lindt
Collaborative Virtual and Augmented Environments
Fraunhofer FIT
Schloss Birlinghoven
DE-5374 Sankt Augustin - Germany
+49-2241-142148

richard.wetzel@fit.fraunhofer.de,
irma.lindt@fit.fraunhofer.de

Annika Waern, Staffan Johnson
Game
Interactive Institute
Box 1197
SE-164 26 Kista - Sweden
+46-8633-1618

annika@sics.se,
staffanj@tii.se

ABSTRACT

Mixed Reality games are becoming more and more popular these days and offer unique experiences to the players. However, development of such games typically still requires expert knowledge and access to Mixed Reality toolkits or frameworks. In this paper, we present the so-called Magic Lens Box that follows a different approach. Based on standard hardware The Magic Lens Box enables game designers with little technological background to create their own Mixed Reality games in a simple yet powerful fashion. We further outline the development process of the magic Lens Box, describe the conceptual model behind it and discuss three games that have been developed with our system. Evaluation of these games shows the viability of our approach, enabling the creation of a variety of rather different Mixed reality games while keeping the development process simple.

Categories and Subject Descriptors

H.5.1 [**Multimedia Information Systems**]: Artificial, augmented, and virtual realities. K.8.0 [**Personal Computing**]: General - Games.

General Terms

Design, Human Factors, Languages.

Keywords

Mixed Reality Games, Mixed Reality, Augmented Reality, Game Authoring, Game Development.

1. INTRODUCTION

Mixed Reality (MR) combines the real environment with a virtual scene to a coherent, interactive experience [1] [14]. While MR is used for a variety of applications including architectural walkthrough, design reviews, and training [4] [5], MR also opens up new design possibilities for games: In MR games, users are able to see their real environment, e.g. their home town or their office extended with virtual game artifacts allowing them to experience familiar environments in a new way. Secondly, in MR games users are not represented by an avatar any more as in traditional computer or console games but represent themselves potentially leading to an immersive experience. And finally, in MR games, users move in the game world by moving in the real environment, providing an intuitive and fascinating way to interact and navigate in the game world. Seen this way, MR games are pervasive games in the sense of Montola [15].

While MR games have a large potential regarding new and immersive game experiences, the development of such games is still in its infancy. Current MR games are mostly developed based on MR toolkits or frameworks which require expert knowledge and often highly specialized and thus difficult to assemble MR hardware systems (see section 2 for an overview of existing games). This prevents non-MR experts like game designers from using MR technology for their games.

In this paper we will present the Magic Lens Box which is our approach to MR game development. The Magic Lens Box consists of off-the-shelf hardware, a conceptual model and a respective authoring environment to specify MR games. The Magic Lens Box is targeting non-MR experts and therefore contains everything needed to develop MR games. In the paper we will describe the development of the box in detail and will analyze three sample games that were realized based on the Magic Lens Box. Based on the analysis of three sample games built with the box, we will argue that the box manages to simplify MR game development on the one side and still supports a wide range of different MR games on the other.

The Magic Lens Box was developed in parallel with the so-called Location Box focusing on GPS and RFID technology and one of our example games was developed using both boxes (http://www.boxed-pervasive-games.org).

The paper is structured as follows: Section 2 discusses related work. Section 3 gives an overview of the Magic Lens Box and its development. The conceptual model that is used to specify MR games is explained in Section 4. Section 5 describes the three games that were realized with the Magic Lens Box and explains their implementation. Section 6 compares and analyses the games in order to see to which extent the Magic Lens Box approach enables the creation of truly different games or gaming experiences before we give an outlook on interesting future research directions.

DIMEA'08, September 10–12, 2008, Athens, Greece.
Copyright 2008 ACM 978-1-60558-248-1/08/09…$5.00.

2. RELATED WORK

When we designed the Magic Lens Box we looked and analyzed a variety of different MR games in order to understand the game experiences as well as the respective MR systems and interaction techniques.

One of the first MR games that were developed was ARQuake [16] – a MR version of the famous first person shooter Quake where the player has to fight monsters and collect game artifacts. The game uses a laptop in a backpack and a head-worn display. Furthermore, the player is equipped with a gun-shaped interaction device. Further MR games based on similar hardware setups followed including Human Pacman [6] and Epidemic Menace [12]. However, there is a clear trend towards hand-held systems for MR games. Hand-held MR systems are based on mobile computers such as ultra-mobile PCs, PDAs or mobile phones and a computer vision-based pose tracking. Examples of hand-held MR games include ARSoccer [17], Invisible Train [18], Virtuoso [19] and ARTennis [9]. For the development of the Magic Lens Box we decided to use a hand-held MR system, since a variety of different game experiences can be delivered via such systems, since the hardware is standard or mostly standard and the potential users might already posses the required hardware system.

The idea to develop an authoring system for MR games was inspired by approaches for user-created content that were quite successful in raising the interest of a large community such as Geocaching [22], where users create quizzes for others based on GPS technology, or Lego mindstorms [21], where users can script and define the behavior of Lego robots, and finally “modding” (modification) tools that are provided with some computer games [11]. In contrast to MR authoring systems such as MARS [8], AMIRE [7] or DART [13], where the application type is not limited, we believe that it is important to design MR authoring tools for specific applications (e.g. games) in order to allow non-MR experts to create their own MR experiences.

3. DESIGN PROCESS

This section gives an overview of the design of the Magic Lens Box. It lists and explains the main objectives, describes the iterative design approach as well as the implementation and the interaction techniques.

3.1 Objectives

MR technology is difficult to handle due to missing supporting tools, unreliable hardware and further issues that are still subject to research (e.g. precise and unrestricted tracking). For the Magic Lens Box our main objectives were to simplify the development of MR games but yet to support a wide variety of different and interesting gaming experiences.

Regarding the simplification, we mean that the effort to author and specify a game, the effort to set-up a game, as well as the effort to stage (or restage) a game should be kept low. Furthermore, non-MR experts should be able to use the Magic Lens Box meaning that no MR expert knowledge is required.

Regarding the flexibility, the Magic Lens Box should not restrict the game genre, meaning that it should be possible to develop games of different game genres such as adventure games, jump and run games, or puzzles based on the Magic Lens Box. Furthermore, the Magic Lens Box should support games of different duration, competitive games, replayability, story-driven games, casual as well as action games.

Our two main objectives are somehow contradictory, since supporting a high flexibility in the design makes a system rather more complex and does not simplify it. To achieve the objectives stated above we had to cut down and limit certain aspects. During the iterative design we decided on a certain MR hardware system as well as on a certain set of interaction techniques that are the same for all the games based on the Magic Lens Box.

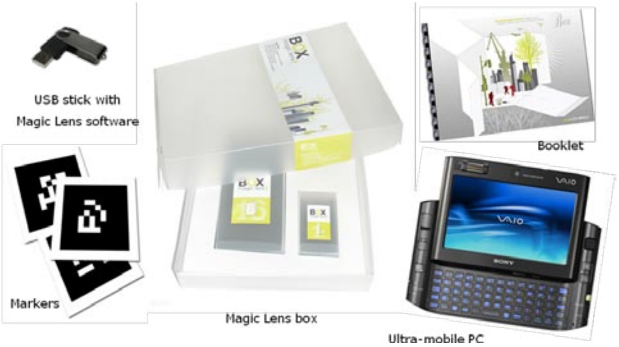

Figure 1. The Magic Lens Box (center) and its content (left and right)

3.2 Design Process

For the design of the Magic Lens Box we worked together with two focus groups: 1) A group of game design students at the University of Düsseldorf (MD.H), Germany. This group, aged 19–24, represents semi-professional game designers. 2) A group of live-action role playing hobbyists in Stockholm, Sweden. This group, aged 20–25, represents semi-professional pervasive game organizers.

We had three feedback rounds with the focus groups. In the first meeting, we showed them the hardware system and the basic interaction techniques. In the second meeting, we introduced them to the Game Creator system, a web-based platform to describe games for the Magic Lens Box and the Location Box mentioned in the introduction (http://www.gamecreator.se). And finally, we sent the Magic Lens Box to the focus groups for a trial period of several weeks and interviewed the participants afterwards.

Initially, the focus group participants were rather hesitant to use the ultra-mobile as a gaming device. They said that the display is small and that the keyboard seems to be cumbersome to use. However when we showed them the usage of the ultra-mobile PC as a “magic lens” their opinion changed and immediately they had several ideas how the ultra-mobile PC could be used in games.

The “magic lens” interaction [2] is a common interaction technique in hand-held MR applications. In this interaction technique, the display shows a video stream of a camera mounted in the back of the display. Users move the display in order to see the real environment overlaid with virtual artifacts. The interaction technique is called “magic lens” since the usage is very similar to the use of an optical hand-held lens.

In the second meeting, we received a lot of feedback from the participants regarding the usefulness and the usability of Game Creator. We realized that the conceptual model that we use to describe games and the Game Creator itself require more explanation. As a result, we wrote and included a step-by-step tutorial for Game Creator and the Magic Lens Box.

During the trial period of several weeks, we observed that the Game Creator and the Magic Lens Box are powerful tools to create different MR game experiences. The results however suffered a bit from some usability issues mainly in the Game Creator web interface, which made us decide to give the Magic Lens Box to another student group as part of a course. Here, the workflow went much smoother, and the students succeeded in implementing the game "The Treasure" as described in section 5.3. As a thorough discussion of Game Creator would go beyond the scope of this paper, we will keep focusing on the underlying model of the Magic Lens Box when we talk about the three sample games that have been implemented so far with it.

The final version of the Magic Lens Box is a transparent box that includes an ultra-mobile PC, a sample game, several printed markers for pose tracking, an USB stick with the Magic Lens Box application and additional markers, as well as a booklet that describes the development of new games using the Game Creator and the Magic Lens Box. Figure 1 illustrates the content of the Magic Lens Box.

3.3 Implementation

The games of the Magic Lens Box (and the games of another box, the Location box) are specified in the web-based box authoring system Game Creator. To run a game on the ultra-mobile PC of the Magic Lens Box, the game needs to be downloaded by running a Python script. The game is then stored on the local computer as a set of XML files and a set of media files. The downloaded data is interpreted by an application running on the ultra-mobile PC. The box application is implemented in C++. The application uses the MR viewer Marvin [3] to display the 3D models on top of a video image. Furthermore, it uses the computer-vision tracking library ARToolkitPlus for marker tracking [20], an extension of the original ARToolkit [10], specifically designed for mobile devices. On start-up, the box application reads in the downloaded game description files and initializes the game accordingly.

During run-time of a game, the local game data can be synchronized with Game Creator, e.g. to update several ultra-mobile PCs running the same game. The run-time synchronization works as follows: If an internet connection is available, e.g. either WiFi or GPRS, the box application pulls all properties that are used within a given game from Game Creator via a HTTP request for. The game is identified via a unique string (so called API key). Subsequently, for all properties used in the game, the respective values are queried from Game Creator. Game Creator delivers the property values as XML file. The Magic Lens Box application parses the XML file and updates the local game state accordingly.

3.4 User Interaction

User interaction was apart from the system hardware another aspect that we limited in order to simplify the Magic Lens Box for game developers. The interaction techniques that are supported by the Magic Lens Box are: 1) Looking with the camera of the ultra-mobile PC at a single marker; 2) Looking with the camera of the ultra-mobile PC at two markers.

The interaction techniques work as follows: When looking at a single marker, one or several 3D models, sound or other media will be displayed to the user. When looking at two markers, some action will be triggered that typically implies changes in the game state, e.g. a game artifact might be collected or applied in the game. Supporting only two interaction techniques seems quite limiting. However, as we will see in section 5 the two interaction techniques are generic enough to be usable and useful in a variety of MR games.

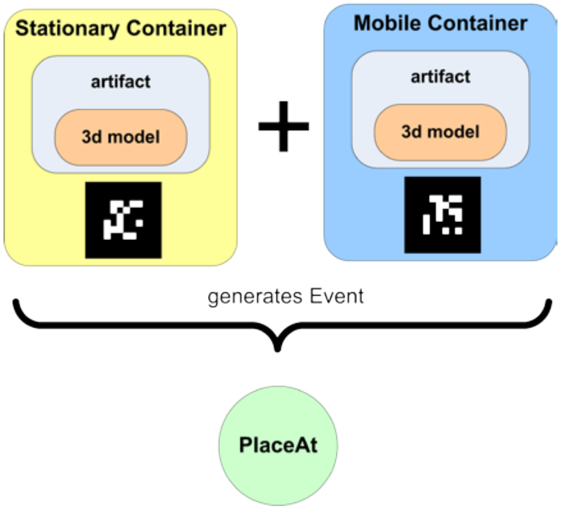

Figure 2. The basic interaction of the Magic Lens Box: Registering two markers at the same time by the UMPC generates a PlaceAt event.

4. THE MAGIC LENS BOX

In order to reach our objectives (see section 3.1), it was important to create an uncomplicated yet flexible enough model to describe games. Game designers should be able to grasp it easily and not be (too) limited in their choices. We started by looking for a simple interaction method that still enabled a variety of different games. We decided to make use of something we would later call a *PlaceAt event*. This activity consists of bringing two ARToolkitPlus markers together as illustrated in Figure 2. Sample games we could think of utilizing this mechanism included a butterfly hunt (catching butterflies in a net), an Augmented Reality Tamagotchi (feeding the character and providing it with toys) and a murder mystery (analyzing clues with different tools). We then analyzed these games more thoroughly and what other aspects they had in common and based our first conceptual model on it. This model underwent some changes during implementation but stayed true to the original throughout the development.

4.1 Game Objects

In the Magic Lens Box we distinguish between three different types of game objects: *stationary containers*, *mobile containers* and *artifacts*.

Stationary containers are meant to not change their actual positions in the real world. They are unmovable aspects of the game. Think of them as locations you can visit, scenery or backdrops. *Stationary containers* can hold *artifacts* and are always connected to *markers*. They can also store string and integer values in user-defined *properties*.

As their name implies, *mobile containers* can be moved around in the real world. The players can pick them up, use and collect them. Other names could be tools, equipment, treasures or items. *Mobile containers* can hold *artifacts* and are always connected to *markers*. They can also store string and integer values in user-defined *properties*.

Artifacts can be associated to *stationary* and *mobile containers*, and this association can be changed during the game. *Artifacts* are never directly represented with *markers*, but can be connected to *3d models*, and they can also store string and integer values in user-defined *properties*.

The characteristics of the different game objects typically create the following hierarchy: A *stationary* (or *mobile*) *container* is connected to a *marker* and holds an *artifact*. The *artifact* in turn holds a *3d model*. This setup can also be seen in Figure 2, and it manages to keep game logic separate from the visual style of the game. *3d models* can easily be removed from *artifacts* and replaced by others without having to tamper with the other settings of the *artifact* itself.

4.2 Assets

Assets are responsible for input from the user (*markers*) and output to the user (*3d models*, *sounds*, *text*).

The user has a choice of 4096 pre-defined *markers* which should be sufficient for all games. The size of the *marker* can be specified, and each *marker* can be attached to a *stationary* or *mobile container*. This also defines whether a marker will later be fixed to a certain position in the real world, or if players will be able and expected to pick it up and carry it around.

For *3d models* the Magic Lens Box supports the following 3d formats: OBJ, VRML, X3D and Collada including animations, sounds and videos. *3d models* can be attached to *artifacts*.

Aside from sounds that are a part of a *3d model*, they are also supported independently. These *sounds* can be triggered via *actions*.

As *sounds*, *text* can be specified and then displayed with the corresponding *action*.

4.3 Rules

Rules combine *events*, *conditions* and *actions*. They are also attached to one or several *game objects*. A *rule* is processed whenever the expected *event* from the attached *game object* is triggered. If all *conditions* that are part of the *rule* are true, all *actions* are executed.

4.3.1 Events

Events are triggered by *game objects*. Two of them can be directly invoked by the user (*lookAt* and *placeAt*) while the other (*propertyChanged*) typically is a result of an *action* performed by another *rule*.

The *lookAt event* is triggered every time the player points with the camera at a *marker* and the *marker* is fully registered.

Whenever a *marker* associated to a *mobile container* is placed next to a *marker* associated to a *stationary marker* (and both are registered by the camera) the *placeAt event* is triggered.

Every *action* resulting in the change of a *property* value automatically triggers a *propertyChanged event*.

4.3.2 Conditions

With *conditions* the game designer can further specify whether the actions in a rule should be executed or not.

The *containerHasArtifact condition* checks if a certain *container* is currently connected to a certain *artifact*.

With the *propertyHasValue condition*, *properties* can be compared to fixed values or other *properties* (even from other *game objects*). It can be checked if the first value is equal, bigger, bigger or equal, smaller, smaller or equal to the second value.

4.3.3 Actions

If all *conditions* are true, then the *actions* are executed in order of specification.

The *attachArtifact* action connects any *artifact* to a *stationary* or *mobile container* whereas the *removeArtifact action* disconnects any *artifact* from a *stationary* or *mobile container*.

With the *changeProperty* action, *property* values of any *game object* can be adjusted. They can be set to a certain value, or another value can be added or subtracted from it. These other values can either be fixed or be coming from other *properties*.

Sound files can be played once with the *playSound action*, and pre-defined text is shown to the player with the *showText action*.

4.4 Sample Code

The following code snippet gives an example of how the xml code generated by Game Creator looks like. A *mobile container* with a *property* that keeps track of the player's gold. Additionally, there is a *stationary container*. Every time the player looks at the *stationary container*, the gold is increased by 100.

```
<mobile-container>
  <id>1</id><name>purse</name>
  <property>
    <id>50</id><name>score</name><value>0</value>
  </property>
</mobile-container>

<stationary-container>
  <id>2</id><name>treasure</name>
  <marker><id>10</id></marker>
  <rule><id>102</id>
</stationary-container>

<rule>
  <id>102</id><name>Add 100 points</name>
  <event><type>lookAt</type></event>
  <action><type>changeProperty</type>
          <owner-id>1</owner-id>
          <property-id>50</property-id>
          <operator>+</operator>
          <value>100</value>
  </action>
</rule>
```

5. IMPLEMENTED GAMES

In this part, we will describe three games that have been implemented with the Magic Lens Box: “The Alchemists”, “Interference” and “The Treasure”.

5.1 The Alchemists

5.1.1 Overview

“The Alchemists” was designed as a prototypical example game for the Magic Lens box. The idea was to make it available as a sample game that could be set up and played quickly as a user first got their hands on the box.

In the game, players take on the role of alchemists and try to brew different magical potions. In order to do so they must find ingredients that are hidden in the playing area, put them in their alchemist’s bag and combine them in their cauldron. The exact nature of the potion and its quality is determined after the brewing process is completed. It is possible to brew love, fear and power potions, and all of them need three ingredients, whose exact nature is unknown to the players. The player who first scores a certain number of points for his potions wins the game.

5.1.2 Reception

Player reception was mixed but mostly positive. Some players perceived the game as being too simple: *“It would be nice if you could actually do something with the different potions, like that you need them to solve other tasks in the game.”*

While this criticism certainly is reasonable, “The Alchemists” can easily be extended in such a way with only the addition of some other game objects and a few rules. Furthermore, we intentionally left out such elements as the game was supposed to serve as the sample game that comes with the box and as such it should spark new ideas.

Most players however liked the action-oriented gameplay and they saw much potential for the game: *“I think this is a great game that you can play to get to know other people. I can see this played on school trips or during corporate events.”*

5.1.3 Development

“The Alchemists” was the game we developed while also developing the Magic Lens Box. The game was a good test-bed, as it made heavy use of the main interaction technique of putting two markers together as well as also demanding basic mathematical functions to calculate the value of the potions.

The player has to bring the bag (*mobile container*) close to an ingredient (*artifact* attached to a *stationary container*) to pick it up (*attachArtifact*). The ingredient can then be dropped again into the cauldron (*stationary container*, *removeArtifact*).

Every time an ingredient is dropped into the cauldron, the cauldron keeps track of the “potion values” of an ingredient in three different properties (*changeProperty*). These values can be -1, 0 or 1 depending on the suitability of the ingredient for each of the possible potions and are added up for all ingredients put into the cauldron.

As soon as three ingredients have been put into the cauldron (*rule* triggered by *propertyChanged event* with *propertyHasValue condition*), it is decided which kind of potion was brewed after all by comparing the “potion values” (*several propertyHasValue conditions*).

Especially the scoring process demanded that you could attach a *rule* to several *game objects* at the same time, something we had overlooked in the first design process for the Magic Lens Box and quickly implemented. It also proved very useful that instead of having to specify a concrete *game object* or *property* inside a *condition* or *action* of a *rule*, you could always instead automatically use the *game object* or *property* that invoked the *rule*.

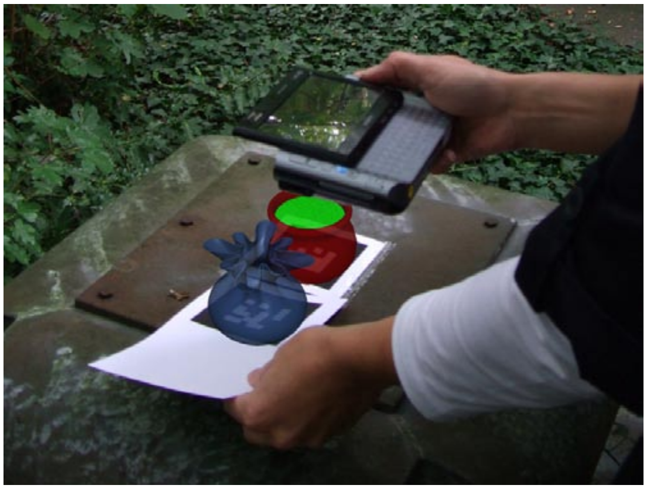

Figure 3. The Alchemists: A player puts an ingredient from the bag into the cauldron by using the basic interaction of the Magic Lens Box.

With the finished Magic Lens Box, the (re)creation of “The Alchemists” can be done rather quickly with most time being used up by specifying markers and uploading 3d models to the Game Creator server. The pure game logic however can be represented in only a handful of generic rules. At the same time it is very easy to adjust the game to one’s needs, e.g. adding more ingredients or potions to the mix or giving the players something to do with the potions as requested by some players.

5.2 Interference

5.2.1 Overview

“Interference” (http://www.interference.nu) is a game that combines the Magic Lens Box with the Location Box to create a unique gaming experience. It was conceived as showcasing different types of playing styles in the very same game to illustrate a variety of different aspects of a typical pervasive game.

The game is played by a group of six to eight players for three to four hours. The players take roles as telecommunication specialists asked to investigate disturbances in the local network. At first they map out this network with the help of the Magic Lens Box technology. They need to follow virtual power lines to central nodes only to discover another mysterious, hidden network only to be drawn into a power struggle between three characters played by actors. The players encounter hidden memories, magical rituals and in the end must choose between two uncertain endings with political undertones (anarchy or complete control).

5.2.2 Reception

“Interference” has so far been staged four times in Kista, Sweden, and three times in Düsseldorf, Germany, during the months January and February 2008.

Player comments after playing the game were generally positive. They liked the many different aspects of the game and the combination of modern technology, actors and playing outside in the real world. However not all of them were able to follow the rather complex storyline. Playing outside in winter for several hours caused some additional problems of both practical and technical nature.

The Magic Lens Box part of the game got positive reviews with one player commenting: *“You got a feeling that you were connected to something bigger. You could do the same with compass, but you wouldn’t get the same feeling.”*

Additionally the sleek design of the UMPCs was seen as really fitting the setting of the initial game scenes. In some groups mapping out the network became a group effort with different players stating their opinion on where to go next, while in other groups one person with good orientation skills took the lead.

Figure 4. Interference: A player using the UMPC to map out the virtual network.

5.2.3 Development

The main implementation for “Interference” was done by the development team of the Magic Lens Box which therefore already had quite some experience with the system. However, one very interesting aspect was the fact that the game had to be restaged at a totally different location. This called for adjusting the game to the new location as the virtual network has to be geographically placed to take local characteristics into consideration, e.g. street layout, interesting or atmospheric areas and the venue for the final scene. From a development point of view, the only things that needed to be changed were the *3d models* of the networks. Before, a network node might have had three power lines going through it, while now there might just be two, also pointing at different directions. The newly added *3d models* then only had to be attached to the already existing *artifacts*, making sure that the game logic stayed untouched.

Unlike in the other two games, in “Interference” the UMPCs are connected to the game server and constantly check if the game state has changed as described at the end of chapter 3.3. This functionality was added to the Magic Lens Box to cater for affordances like this, where the game state was modified not by the player himself.

The actual player interaction was implemented very easily. An “open” gate was represented by an *artifact*, and a “closed” gate by another. When the “status” *property* was changed by means of the Location Box, a rule was invoked by a *propertyChanged event*. The rule then modified the *stationary container* the players were inspecting by exchanging the attached *artifact* (*removeArtifact*, *attachArtifact*).

5.3 The Treasure

5.3.1 Overview

“The Treasure” was implemented by a group of three students over the course of one semester during a Game Design Lab at the Bonn-Aachen International Center for Information Technology (B-IT). The students wanted to focus on the game design aspect when creating their first Mixed Reality game which suited the concept of the Magic Lens Box extremely well.

In the game, one group of players tries to find a treasure hidden away by the former ruler over Castle Birlinghoven. On their way, the players have to solve riddles and tasks given to them orally by the real statues scattered around the castle grounds. One of these statues can be seen in Figure 5.

5.3.2 Reception

As of now, “The Treasure” has only been staged once in the spring of 2008. Players were very engaged in the game which could be observed by the fact that they were running at several times during the game although there was no time limit enforced in the game.

The difficulty level of the riddles seemed well balanced as most of them were not solved by the players immediately, but neither did they get stuck for a too long time.

Throughout the game players were not quite sure which markers they were supposed to take with them. In one instance they even ripped off a stationary one that was thought to be securely taped to a table.

Players were especially fond of the clever use of the statues in the game and named this as their favorite feature: *“It was really great to be able to hear the old statues talking, this brought me really into the mood that we were searching for an old treasure.”*

5.3.3 Development

Unlike the other two games, “The Treasure” was not developed by somebody experienced with the Magic Lens Box. Furthermore, the students also had no previous experience in creating Mixed Reality games.

In the beginning the students were shown “The Alchemists” in order to get a feel for what Mixed Reality games could be. Afterwards they were given ample time to come up with a game concept. During this stage it was important for us not to tell the students anything about the Magic Lens Box. This insured they were not constrained or influenced in their ideas by the technology. After they toyed around with several ideas, they finally settled on "The Treasure" as they were very interested to tailor the game to Campus Birlinghoven to create a story-rich MR game.

At this point we gave them an overview of the Magic Lens Box, and let them ponder about the best way to implement their game. In the beginning the students had some problems understanding the difference between *mobile* and *stationary containers*. After getting examples, however, on how to use them best, the students quickly noticed that all the characters in the game (real statues) were best represented by *stationary containers*, while the items the players would have to use for solving the riddles became *mobile containers* (as these would be carried around by the players).

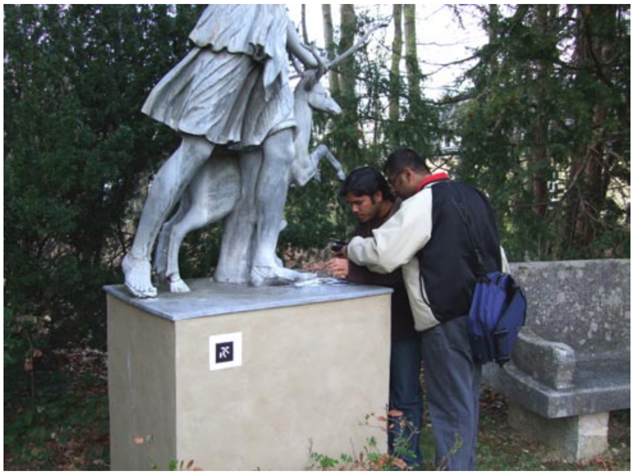

Figure 5. The Treasure: Two players getting a new quest from a statue.

The students then sketched out all their *game objects* and also started to produce *sound* files and *3d models* for the game. As this content is separated from the game logic in the Magic Lens Box, the students were able to work on both at the same time which greatly sped up the development process.

The interaction within the game is mostly based on the *placeAt event* responsible for deciding if the riddle was solved by bringing the correct item (*containerHasArtifact condition*). Additionally, the *lookAt event* and *conditions* checking progress the players have made (*propertyHasValue*) are used for making the statues talk to the players (*playSound action*).

At the end of the semester, the students staged their game and were very satisfied with the positive feedback they got from the players. In their opinion, working with the Magic Lens Box had a great effect on their success:

"I personally liked this way to create the game, mostly because it was really feasible to create an Augmented Reality game at this short time period we had. If we would have to develop our own functionality it would not be possible at all, but with Game Creator and the Magic Lens Box we could just focus on the game design." (comment by one of the students)

6. CONCLUSIONS

When comparing the three games that have been implemented so far, we can see that they differ in quite many aspects (see Table 1). With the Magic Lens Box, we were able to create competitive games as well as cooperative ones, games that lean more towards action and others that are much more story oriented. Although the games all share the same set of interaction techniques, the resulting gaming experiences are all quite diverse. Whereas "The Alchemists" makes people run around the playing area trying to be the first to brew a certain potion, in "The Treasure" players take on a more cautious approach eager not to miss any slight hint and always contemplating on the riddles given to them. "Interference" on the other hand draws the players into a struggle of unknown proportions and demands role-playing and interaction with non-player characters.

The games however also share some characteristics: They are all very easy to set-up and to restage elsewhere. A game master only needs to print out the markers and place them in the playing area. The latter is a little bit more complicated for "Interference" and "The Treasure" as the markers need to be put at pre-planned locations.

Table 1. Comparison of distinguishing game characteristics.

Characteristic	**The Alchemists**	**Interference**	**The Treasure**
Genre	Scavenger Hunt	Urban Adventure	Riddle Solving
Playing time	30 minutes	4 hours	1 hour
Competition	2 teams competing	1 team, cooperative	1 team, cooperative
Actors	No	yes	no
Story-dependence	Low	Very High	High
Casualness	High	Low	Medium
Action-Orientation	High	Low	Medium
Effort needed for set-up	Very Low	Low	Low
Effort for restaging game elsewhere	Very Low	Low	Very Low (if statues are available)

As of now the system already supports a variety of different games, and the reception of all of them has been positive. This makes us confident to say, that the Magic Lens Box enables the creation of diverse and fun Mixed Reality games

In the future, the range of games can certainly be increased further by the simple addition of more events, conditions and actions. One example could be the inclusion of a timer event. The control for text and sound are only rudimentary at the moment and offer ample room for improvement.

So far the main criticisms of the Magic Lens Box were general usability issues, for example with Game Creator: *"I think the main reason for these difficulties were the UI. For example, as soon as I got familiar with the way Game Creator functions I did not have big problems with implementing the game."* (Comment by one of the B-IT students developing "The Treasure")

Our own experiences with the Magic Lens Box and especially this kind of feedback from the student group that created "The Treasure", however, make us believe that our approach simplifies the creation and modification of MR games, even for inexperienced users. The game mechanics we supply are sufficient enough, so developers can focus on the design of the game instead of implementation issues.

In our opinion it is the pervasive character of Mixed Reality games that inspires much richer gaming possibilities without the need for an elaborate (and thus complicated) toolset. The real world offers many opportunities to flesh out the games and create unique and rewarding experiences: Ingredients hidden in dark cellars, talking statues and surprising plot developments through interference by actors. The Magic Lens Box enables game designers to create such games easily and reliably.

7. ACKNOWLEDGMENTS

The authors wish to thank Michael Wittkämper for his contributions to the development of the Magic Lens Box, the B-IT students Janna Brener, Melissa Sudjono and Hagop Araklian for working with the box and creating “The Treasure”, Sabiha Ghellal and Nina Dautzenberg for the professional graphics design of the Magic Lens Box, all members of the IPerG project for work, dedication, comments and discussion, and the European Commision for funding of IPerG (FP6-2003-IST-3-004457) as part of the 6th Framework.

8. REFERENCES

[1] Azuma, R.: A Survey of Augmented Reality. Presence: Teleoperators and Virtual Environments 6, 4 (August 1997) 355 – 385

[2] Bier, E., et al., "Toolglass and Magic Lenses: The See-Through Interface," *Proc. ACM Conf. Computer Graphics and Interactive Techniques (Proc. Siggraph),* ACM Press, 1993, pp. 73–80.

[3] Broll, W., Lindt, I., Ohlenburg, J., Herbst, I., Wittkämper, M., and Novotny, T.: An Infrastructure for Realizing Custom-Tailored Augmented Reality User Interfaces, IEEE Transactions on Visualization and Computer Graphics, 11 (6), Nov. 2005, 722-733.

[4] Broll, W., Lindt, I., Ohlenburg, J., et. al. Arthur: A Collaborative Augmented Environment for Architectural Design and Urban Planning. Journal of Virtual Reality and Broadcasting, 1(1), December 2004. ISSN 1860-2037.

[5] Brooks Jr., F. What’ s real about Virtual Reality? In VR ’99: Proceedings of the IEEE Virtual Reality, pages 2–3, Washington, DC, USA, 1999. IEEE Computer Society Press.

[6] Cheok, A., Goh, K. H., Liu, W., Farbiz, F., Fong, S. W., Teo, S. L., Li, Y., and Yang, X. 2004. Human Pacman: a mobile, wide-area entertainment system based on physical, social, and ubiquitous computing. Personal Ubiquitous Comput. 8, 2 (May. 2004), 71-81.

[7] Doerner, R., Geiger, C., Haller, M., Paelke, V. Authoring Mixed Reality. A Component and Framework-Based Approach. In First International Workshop on Entertainment Computing (IWEC 2002), May 2002.

[8] Gueven, S., Feiner, S.. Authoring 3D Hypermedia for Wearable Augmented and Virtual Reality. In Proceedings of ISWC’03, pages 118–126, 2003.

[9] Henrysson, A., Billinghurst, M., Ollila, M.,. Face to Face Collaborative AR on Mobile Phones. In Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR 2005), October 5th – 8th, 2005, Vienna, Austria.

[10] Kato, H. and Billinghurst, M. 1999. Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System. In Proceedings of the 2nd IEEE and ACM international Workshop on Augmented Reality (October 20 - 21, 1999). IWAR. IEEE Computer Society, Washington, DC, 85.

[11] Kücklich, J. Precarious Playbour: Modders and the Digital Games Industry. In Fibreculture Journal, issn 1449 – 1443, issue 5(2005), Australia.

[12] Lindt, I., Ohlenburg, J., Pankoke-Babatz, U., and Ghellal, S.: A Report on the Crossmedia Game Epidemic Menace. ACM Computers in Entertainment (CIE) volume 5, issue 1 (Jan. 2007), 2007.

[13] MacIntyre, B., Gandy, M., Bolter, J., Dow, S. and Hannigan, B. "DART: The Designer’s Augmented Reality Toolkit", In Proc. of IEEE and ACM International Symposium on Mixed and Augmented Reality, Tokyo, Japan, 2003, pp. 329-330.

[14] Milgram, P., Takemura, H., Utsumi, A. and Kishino, F.: Augmented Reality: A class of displays on the reality-virtuality continuum Augmented Reality: A class of displays on the reality-virtuality continuum. (pdf), SPIE Vol. 2351-34, Proceedings of Telemanipulator and Telepresence Technologies (1994).

[15] Montola, M. Exploring the Edge of the Magic Circle. Defining Pervasive Games. Proc. Of Digital Experience: Design, Aesthetics, Practice conference, Copenhagen, 2005.

[16] Piekarski, W. and Thomas, B. 2002. ARQuake: the outdoor augmented reality gaming system. Commun. ACM 45, 1 (Jan. 2002), 36-38.

[17] Reimann, C.; Paelke, Volker; Stichling, Dirk: Foot-based mobile Interaction with Games. In: ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE). Singapore, Juni 2004.

[18] Wagner, D., Pintaric, T., Ledermann, F., Schmalstieg, D., 2005. Towards Massively Multi-User Augmented Reality on Handheld Devices. Proceedings of the Third International Conference on Pervasive Computing (Pervasive 2005), Munich, Germany.

[19] Wagner, D., Schmalstieg, D., Billinghurst, M. Handheld AR for Collaborative Edutainment. Proceedings of 16th International Conference on Artificial Reality and Telexistence (ICAT), 2006.

[20] Wagner, D. and Schmalstieg, D.: ARToolKitPlus for Pose Tracking on Mobile Devices. Proceedings of 12th Computer Vision Winter Workshop (CVWW'07), February 2007

[21] Lego.com MINDSTORMS NXT Home, mindstorms.lego.com [Available: 18-05-2008]

[22] Geocaching.de - Das Geocaching-Portal, geocaching.de [Available: 18-05-2008]