# Tangible Audio Game Development Kit: Prototyping Audio Games with a Tangible Editor

**Michael Urbanek**
TU Wien
Institute for Design &
Assessment of Technology
Vienna, A-1040, Austria
michael.urbanek@tuwien.ac.at

**Florian Güldenpfennig**
TU Wien
Institute for Design &
Assessment of Technology
Vienna, A-1040, Austria
florian.gueldenpfennig@tuwien.ac.at


*TEI 2017, March 20–23, 2017, Yokohama, Japan.*
ACM ISBN 978-1-4503-4676-4/17/03.
http://dx.doi.org/10.1145/3024969.3025077

## Abstract
Audio games are a genre of computer games that waive visual output and solely rely on audio. The majority of audio games (as well as video games) lack publicly available editors that allow the creation of new games or maps within the framework of the respective game. Additionally, no existing game or map editors make use of tangible interaction to facilitate embodied interaction during the creation of games. However, many elements in audio games can be mapped in a profitable way to a tangible editor. We therefore propose the Tangible Audio Game Development Kit that allows fast, iterative and collaborative audio game prototyping by combining physical objects. In this paper, we introduce a first prototype of this concept and present a user study featuring three game design experts. We discuss the results and explain the next steps in advancing the audio game editor based on the feedback.

## Author Keywords
Audio Games; Tangible User Interface; Game Editor; Prototyping.

## ACM Classification Keywords
H.5.2 [Information interfaces and presentation (e.g., HCI)]: User Interfaces.

For a video demo please visit https://youtu.be/FoKsIGo32xo, last accessed on Dec. 22, 2016.

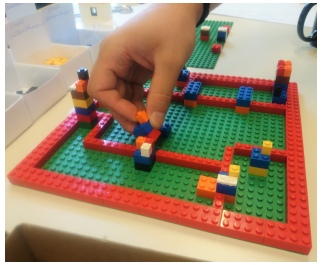

**Figure 1:** One of the experts (participant 2) playing the pre-built game. The expert explained the game while he played through it by moving his avatar. In this picture, he removed a destroyed obstacle.

## Introduction & Related Work

Over the last decades, the video games industry has turned into a multi-billion Dollar business. While many of the current games rely on bespoke computer graphics, other productions achieved major successes by focusing on excellent game play or game narrative (e.g., Minecraft, which implemented few graphical features as a deliberate design decision). Besides the mainstream video games, there is a small but active niche-community, which waives visual output completely: audio games. The audio gamer does not want or is not able to rely on visual feedback (e.g. visually impaired people) and therefore solely relies on audio. Generally, such games are developed by people without visual impairments for people without (full) vision. However, as the positive response to the commercial iOS audio game series Papa Sangre [1] shows, a great number of people with normal vision enjoys audio games as well. Such audio games can be played in front of a computer [7], on a mobile phone [1], employing in- or out-door user tracking [5, 10] or with an extra utility (controller) [3]. Usually, these games are first person-centered experiences using spatialized 3D sound that allows sound localization, creating a space that can be explored through audio. For better localization and immersion, headphones are recommended. In audio games, every object that does not emit or reflect sound is "inaudible and not perceivable" [6]. In analogy to video games, such inaudible objects would be invisible.

*Creating games*: Few video games (e.g. Warcraft III, 2002) provide a map editor in addition to the base game that enable game designers, prosumers (e.g. modders) or casual players to extend the original game. The arrangement of objects (by dragging/dropping of, e.g., characters, walls or trees) with the editor specifies the look and setup of the resulting new level. Hence, the editor including objects can be considered an editable representation of the game itself.

In addition to placing objects, some editors support the users in scripting for introducing logic to the game and therefore in creating their own game mechanics. Usually, game editors run on a computer and are operated by classic I/O interfaces only (e.g. keyboard, mouse). This year, a tangible video game creation kit named Boxels (http://www.bloxelsbuilder.com/, last accessed on December 22, 2016) was introduced to the market, which makes use of a tablet app to take pictures of physical blocks in different colors. These images are then translated into game levels to create 2D video games. Hence, Bloxels constitutes a commercial tangible editor for designing, for example, 2D Jump'n'Run games.

We argue that the concept of a tangible editor is also very appropriate for audio games for two reasons. Firstly, due to the nature of audio games (no graphics, less complexity, focus on strong narrative) they constitute a promising candidate for being shaped by tangibles rather than by a software editor. Secondly, using a tangible editor, audio games can be (co-)designed by people with impaired vision by grasping the different objects.

The use of tangibles as well as the combination of them was utilized before in the area of sound design. AudioCubes [8] is a tangible user interface that consists of several cubes that produce and modify sound depending on the spatial relation of the cubes. Another example is the tabletop user interface Reactable [2] that also uses physical artifacts to generate and modify sound. Here, every artifact (cube) also has its own function. The tangible sequencer presented in [4] also enables sound manipulation by incorporating the combination of physical artifacts. The mapping of physical artifacts as utilized in AudioCubes, Reactable or Block Jam has not been employed yet to create a tangible editor for audio games and will act as the primary concept of the Tan-

gible Audio Game Development Kit (TAGDK). TAGDK as proposed in this paper is a mixed spatial relational system that draws on well established tangible interaction principles as proposed by Ullmer and Iishi [9] (physical representation at this prototyping stage: Lego; digital representation: Unity3D).

We therefore propose the concept and first prototype implementation of the TAGDK that allows audio game prototyping by placing and combining tangible artifacts that provide a physical representation of game objects. We introduce TAGDK and test a first implementation with three game design experts (Figure 1 shows one expert interacting with it). Our primary motivation in this paper is to find out if the concept is feasible and to gain further inspiration. We report this feedback and explain how we will use it to advance the TAGDK. In future design iterations we will refine the concept as well as implement a tracking device to remove the *Wizard-of-Oz* component (see User Study section) in this paper (e.g., see Figure 6).

**Figure 2:** The TAGDK elements represented by respective LEGO bricks. Currently, the editor supports 13 different elements from three different categories (*Objects*, *Properties*, *Links*). From back to front: The TAGDK Objects (Avatar, Obstacle, Soundsource, Item, Character, Portal), the TAGDK Properties (Message, Direction, Destructible, Harmful, Move, Win Condition) and TAGDK Links.

## TAGDK: Tangible Audio Game Development Kit

The TAGDK allows audio game prototyping by placing and combining haptic artifacts. The motivation of this framework is that even people with little training can intuitively pick TAGDK artifacts, place them on a table, hence, prototype the audio game, "sit back", and play the game immediately without knowledge of software programming. Due to this agility, the TAGDK can be used to iteratively prototype audio game concepts including rapid revisions (see Figure 4 for an illustration). It can also be used to build audio games collaboratively (e.g. by sharing one kit on one table). The main target group is people who are interested in audio game design (beginners and experts) with or without normal vision.

| Element | Description |
|---|---|
| Avatar (O) | First-person avatar controlled by the player. |
| Obstacle (O) | Insurmountable object. |
| Soundsource (O) | Continuous source of sound within the game scene. |
| Item (O) | Item a player can interact with. |
| Character (O) | In-game character that is not controlled by the player. |
| Portal (O) | Player is teleported to associated other Portal element (logically connected with Link element). |
| Message (P) | Text-to-speech message. |
| Move (P) | Makes an object movable. |
| Direction (P) | Determines the direction of an element (e.g. move-axis). |
| Destructible (P) | Property of an object that flags it as destroyable. |
| Harmful (P) | Property of an object that flags it as emitting damage. |
| Win Condition (P) | Property of an object that ends the game if reached. |
| Link (L) | Logically links two or more objects together. |

**Table 1:** Elements of the TAGDK editor represented by LEGO bricks. Currently 13 elements in three categories (O...*Object*, P...*Property*, L...*Link*) are supported. Elements can be combined to accomplish different responses or outcomes. Note Table 2 for further explanations regarding the combinations of the elements.

In TAGDK, every artifact has a defined function that can be combined with other artifacts. In the following, we will call them the TAGDK elements. In a combination of multiple el-

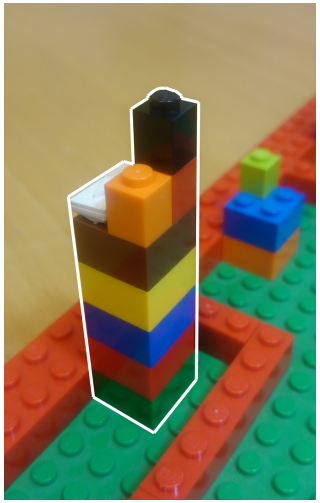

**Figure 3:** Representation of an "enemy" using LEGO bricks. From bottom to top: A Character object (green) that is also an Obstacle (red, so the enemy is not a "ghost") with a Soundsource (blue) that has an Item (yellow) and is Movable (brown). The enemy is Destructible (orange) and Harmful (red), logically Linked (black) with something else and moves in the given Direction (white arrow).

ements, the order is without relevance. In the TAGDK prototype as proposed and tested in this paper (Figure 2), three different categories of elements are introduced: *Objects*, *Properties* and *Links* (see Table 1 for a detailed description). *Objects* include game elements like Avatars, Obstacles, Soundsources, Items, Characters and Portals, which are typically found across different game genres. They also define the main actors or characters of the prospective audio game prototype, similar to the interactable objects (obstacles, portals, interactables) as proposed by Röber and Masuch [6]. *Properties* modify (combined) objects without adding additional complexity. Properties are Message, Move, Direction, Destructible, Harmful and Win Condition. *Links* are used to logically link two or more objects together. The combination space of the TAGDK elements is defined in a 2-dimensional matrix (see Table 2 for an excerpt).

## User Study

The aim of the initial study in this paper is to get feedback from game design experts about the TAGDK. This information will be used to define the design of a tracking device (currently implemented by a *Wizard-of-Oz* component as described below), and to make sure that the proposed game elements are useful, powerful whilst not too complex (are the functions "atomic"?), and clear (are the virtual functions understandable?).

*Experimental prototype:* In the current prototype, we opted for using LEGO bricks as they can be combined and disassembled easily, and more importantly, they were suitable for mapping the required functions of the TAGDK. LEGO bricks are stackable and distinguishable in terms of form, size and color. Every LEGO brick used in this game-expert study constitutes a physical representation of one of the TAGDK elements (cf. also description above): *Objects* are represented by 2x2 sized LEGO bricks, *Properties* by 1x1

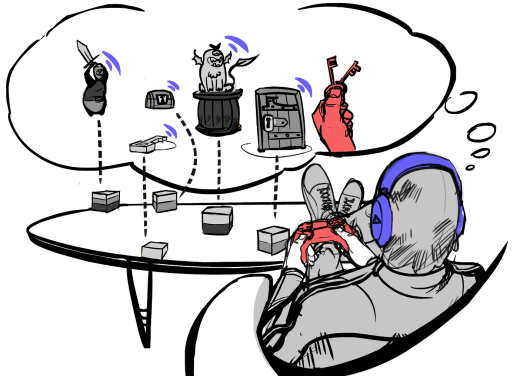

**Figure 4:** The concept of the TAGDK framework illustrated. An audio game designer playing their prototype immediately after placing the artifacts on the table. Every combination of artifacts results in a game object (enemies, items, ...). User input is entered with a wireless controller. User or game feedback is audio only through headphones.

sized blocks (except, the direction element is represented as 3D-printed 1x2 LEGO compatible arrow) and *Links* as 1x1 blocks (see Figure 2). In combination, these LEGO bricks form game objects in an audio game built with the TAGDK. One of the exemplary game objects could be an "enemy" as shown in Figure 3. Note, in the current prototype, the LEGO game map is rendered into a playable audio game by a *Wizard-of-Oz* or researcher, who observes the participant and then uses a Unity3D application that we have created as a tool for editing the respective audio game.

*Study setup:* We tested our TAGDK concept with three game design experts (more than a decade of professional experience each; recruited through the university network) using individual semi-structured interviews and hands-on

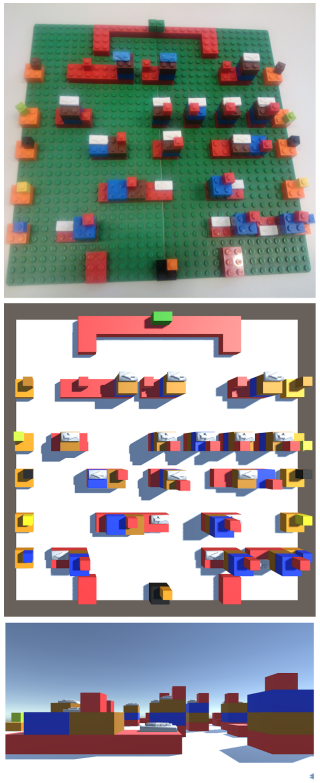

**Figure 5:** A Frogger game built with the TAGDK framework. The first image illustrates the Frogger game built with LEGO bricks by the expert (participant 3). The second image shows the top view of the game, rendered in Unity3D. In the third image, the first-person view of the avatar is shown. This visual output is for debug purposes only and not for the end-user, because audio games waive visuals completely.

| | (A)vatar | (O)bstacle | (S)oundsource | (I)tem | (C)haracter | (P)ortal |
|---|---|---|---|---|---|---|
| (A)vatar | | | Sound while moving | Items to start with | | |
| (O)bstacle | | | Sound when colliding | Drops when destructed (D) | Character perceptible | |
| (S)oundsource | Sound while moving | Sound when colliding | | Item audible | Character audible | Portal audible |
| (I)tem | Items to start with | Drops when destructed (D) | Item audible | | Drops when destructed (D) or at collision (M) | |
| (C)haracter | | Character perceptible | Character audible | Drops when destructed (D) or at collision (M) | | |
| (P)ortal | | | Portal audible | | | |

**Table 2:** An excerpt of the combination possibilities of *Object* elements in the TAGDK framework. A combination of an object with the same object is not supported. The order of combination is without relevance. LEGO brick images taken from https://shop.lego.com/en-US/Pick-a-Brick, last accessed on December 22, 2016.

sessions with the TAGDK prototype. All sessions (lasting 60-90 minutes) were video-recorded for later qualitative thematic analysis (exploratory and inductive coding). During each semi-structured interview, the experts were asked for their critical opinion about the TAGDK and its applicability as an (audio) game editor. The subsequent hands-on sessions were split into two parts. Firstly, the experts were invited to build their own audio game using the TAGDK LEGO prototype. Secondly, a pre-built TAGDK audio game was shown to the experts. The experts then explored this third-party game and explained how they played it. Figure 1 shows one expert while playing the pre-built game.

## First Results & Outlook

The expert interviews provided encouraging feedback on the concept of the TAGDK. All experts understood it with little training time (5-10 minutes) and thought that tangibility might be a promising concept for (audio) game editors due to the extended embodied experience it introduced to the conventionally *flat* software design activity of editing.

"It's a nice tool to test game elements physically. It's much better than describing ideas verbally," participant 1 said. All participants were able to build audio games and explain the pre-built level. Two of the three experts created a Frogger game in which a player had to cross a street without getting hit by an obstacle. The expert's LEGO prototype, the rebuilt Unity3D prototype and the avatar's first person view in Unity3D is shown in Figure 5. Besides approval, the experts also provided valuable feedback for improving the next iteration of the TAGDK.

First of all, the experts commented on the delay between LEGO prototyping and having the playable prototype available due to the *Wizard-of-Oz* component (rendering the game by hand in Unity3D). This critique was anticipated by the authors and the next major effort will be to develop the tracking device as outlined in Figure 6. In this paper, however, we omitted this implementation effort for a hi-fi prototype, as we wanted to obtain the experts' opinion before this

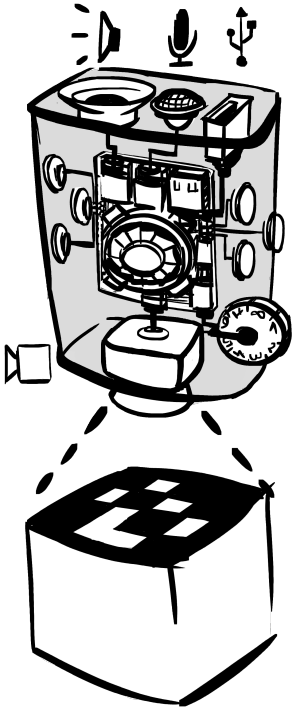

**Figure 6:** Draft of a tangible programmer element for TAGDK (a handheld device with tactile buttons; standalone with no extra PC, etc.). A camera at the bottom identifies game objects with markers. Photographed TAGDK elements then get labeled as different game elements using buttons (instead of different LEGO bricks, it employs different buttons on the device). An integrated speaker provides auditive feedback, a microphone can be used for sound recording (e.g., audio messages for the later game). Additional game sound files can be uploaded through USB.

investment. Their feedback will be considered in the design of this device/hi-fi prototype.

The game design experts thought it might be useful that in a future implementation, individual TAGDK elements should be linkable on an individual basis (i.e., they demanded some extensions of the combination matrix in Table 2). As a specific example, participant 1 wanted to create an object that had to be destroyed to win the game by combining the Win Condition with the Destructible property.

One issue raised by participant 1 and 3 was a missing option for creating templates. In the example of Frogger given above, templates could have been used for the cars without building every car from scratch again. Therefore, the option to create templates will be considered in the next iteration.

In addition, the experts requested more scripting support similar to conventional computer programming (e.g. conditions, or randomization). Still, it was advised to create this function without adding too much complexity.

We also identified issues with using LEGO bricks. The combination of elements led to larger chunks that were not instantly clear with regard to the resulting behavior. The preliminary results showed that the "mental mapping" of the color to its respective function demanded too much time from the designers. Most importantly, we do not want to constrain the concept and its accessibility by relying on color coded bricks, since one important target group of our tangible audio games editor might be visually impaired people. Thus, for the final tangible prototype, LEGO will not be used.

*Outlook:* The next design iteration of this framework will include a tangible prototype, drawing on the expert's feedback about the concept prototype in this paper. We have started experimenting with embedded ODROID computers to build a tangible editor in the shape of a handheld device with tactile buttons (see Figure 6 for description). Game objects will be represented by physical cubes (or other shapes that clearly tell the designer about the respective function) with optical markers. Their properties will be specified by the TAGDK programmer (handheld device) using buttons and other tactile controls. A picture of the whole game area through the built-in camera sets the game objects in relation, providing information about the objects in virtual space. Hence, the handheld and its built-in camera is used to label individual objects (assign functions) as well as to capture the whole map (relations of the objects to each other). With this device we will then conduct further studies with game designers, and we will also use it to evaluate if tangible editors are appropriate for blind designers to create audio games.

We hope that the TAGDK framework raises additional awareness about audio games and encourages more researchers to include non-visual games into their studies.

## Acknowledgments

We would like to thank *Peter Fikar* for creating the illustrations in Figure 4 and Figure 6. His efforts are highly appreciated.

## REFERENCES

1. Somethin' else. 2010. Papa Sangre. (2010). Retrieved October 24, 2016 from *https://itunes.apple.com/en/app/papa-sangre/id407536885?mt=8*.

2. Martin Kaltenbrunner, Sergi Jorda, Gunter Geiger, and Marcos Alonso. 2006. The reacTable*: A Collaborative Musical Instrument. In *15th IEEE International*

*Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*. IEEE, 406–411.

3. Philip Mendels and Joep Frens. 2008. The Audio Adventurer: Design of a Portable Audio Adventure Game. In *Fun and Games*. Springer, 46–58.
4. Henry Newton-Dunn, Hiroaki Nakano, and James Gibson. 2003. Block Jam: A Tangible Interface for Interactive Music. In *Proceedings of the 2003 Conference on New Interfaces for Musical Expression*. National University of Singapore, 170–177.
5. Iana Podkosova, Michael Urbanek, and Hannes Kaufmann. 2016. A Hybrid Sound Model for 3D Audio Games with Real Walking. In *Proceedings of the 29th International Conference on Computer Animation and Social Agents*. ACM, 189–192.
6. Niklas Röber and Maic Masuch. 2004. Interacting With Sound: An Interaction Paradigm for Virtual Auditory Worlds. In *Proceedings of the 10th International Conference on Auditory Display*.
7. Jaime Sánchez and Mauricio Lumbreras. 1999. Virtual Environment Interaction Through 3D Audio by Blind Children. *CyberPsychology & Behavior* 2, 2 (1999), 101–111.
8. Bert Schiettecatte and Jean Vanderdonckt. 2008. AudioCubes: a Distributed Cube Tangible Interface based on Interaction Range for Sound Design. In *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction*. ACM, 3–10.
9. Brygg Ullmer and Hiroshi Ishii. 2000. Emerging Frameworks for Tangible User Interfaces. *IBM systems journal* 39, 3.4 (2000), 915–931.
10. Eric Velleman, Richard van Tol, Sander Huiberts, and Hugo Verwey. 2004. 3D Shooting Games, Multimodal Games, Sound Games and More Working Examples of the Future of Games for the Blind. In *International Conference on Computers for Handicapped Persons*. Springer, 257–263.