This item was submitted to Loughborough's Research Repository by the author.
Items in Figshare are protected by copyright, with all rights reserved, unless otherwise indicated.

# Making the mainstream accessible: what's in a game?

PLEASE CITE THE PUBLISHED VERSION

PUBLISHER

© Springer-Verlag Berlin Heidelberg

VERSION

AM (Accepted Manuscript)

LICENCE

CC BY-NC-ND 4.0

REPOSITORY RECORD

Atkinson, Matthew T., Sabahattin Gucukoglu, Colin H.C. Machin, and Adrian E. Lawrence. 2019. "Making the Mainstream Accessible: What's in a Game?". figshare. https://hdl.handle.net/2134/4461.

This item was submitted to Loughborough's Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

creative commons

COMMONS DEED

**Attribution-NonCommercial-NoDerivs 2.5**

**You are free:**

- to copy, distribute, display, and perform the work

**Under the following conditions:**

**Attribution.** You must attribute the work in the manner specified by the author or licensor.

**Noncommercial.** You may not use this work for commercial purposes.

**No Derivative Works.** You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

**Your fair use and other rights are in no way affected by the above.**

This is a human-readable summary of the Legal Code (the full license).

Disclaimer

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/

# Making the Mainstream Accessible: What's in a Game?*

Matthew T. Atkinson[1], Sabahattin Gucukoglu[2], Colin H. C. Machin[1], and Adrian E. Lawrence[1]

[1] Loughborough University, Loughborough, Leicestershire, LE11 3TU, England
{M.T.Atkinson, C.H.C.Machin, A.E.Lawrence}@lboro.ac.uk
[2] The AGRIP Project, http://www.agrip.org.uk/
sebby@agrip.org.uk

**Abstract.** Though accessible gaming is a well-established phenomenon, few mainstream applications of it exist. We present some of the work of the AGRIP project – an effort to develop techniques to render modern first-person shooter games accessible to the blind and vision-impaired. We discuss some of the low-level accessibility infrastructure employed in the game AudioQuake and compare it to other contemporary research. The project's ultimate goals of generalisation and use of the technology in educational settings are also introduced.

## 1 Introduction

AudioQuake is the first adaption of an existing mainstream game designed specifically for sighted people that has been made playable by blind gamers. It is unique in terms of the range of Internet-enabled gameplay modes it provides. At one level, it could be termed an "accessibility layer" for Quake[1].

This paper describes the work of the AGRIP project – an effort to develop techniques for making mainstream games, tools and their communities accessible to blind and vision-impaired gamers that has been active since May 2003. The approach taken by this project contrasts with other contemporary research [1–3] in the following ways.

**Adaption** — Whilst other projects often develop engine and game platforms anew, this project modifies existing well-designed mainstream technology to improve its accessibility and usability for all. The goal is *not* to retrofit accessibility, but to show how properly-designed systems may be made accessible through the processing and rendering of information at separate levels.

**Generalisation** — An important aim is to use the project to develop a deeper understanding of accessibility barriers that can be used in the development of general techniques to deal with those issues. We hope to create "portable"

* K. Miesenberger et al. (Eds.): ICCHP 2006, LNCS 4061, pp. 380–387, 2006.
© Springer-Verlag Berlin Heidelberg 2006

[1] the seminal first-person shooter from id Software

solutions to accessibility problems, that can be used in other (academic and leisure) settings.

Throughout development, the AGRIP project has been shaped by community feedback. This incorporates that given by users via e-mail and using project mailing lists. Many comments from users and suggestions for improvements have been made. Using the mailing list approach has enabled a number of interesting discussions to take place *between* users of the software, thus giving us a greater insight into how effective the work has been. AudioQuake has been used as the basis of a number of educational workshops at the 2005 International Computer Camp for Vision-Impaired People.

The rest of this paper expands on the user requirements, design, technical and other issues encountered in providing low-level access to the game (essentially local navigation).

## 2 Local Navigation

There are two main strands of navigation present in almost any computer game, and in any problem that involves searching a solution space. These are *global* navigation towards one's ultimate goal – the silver key; the red team's flag; the shortest path between nodes in a tree – and *local* navigation – "How do I get out of this room?". At the low-level game accessibility stage of the project's development, the primary concern was to develop the mechanisms to support effective local navigation.

In this style of game, the player controls their character directly, and is presented with the character's world view. A number of "devices" which act as navigation aids are provided to facilitate accessibility. These devices attempt to resemble real-world mobility aids where possible and encourage the user to navigate the virtual world in a similar way that they do in the real world. For example, to allow players to get a very fine-grained sense of what obstacles/structures surround them, a "sweep" can be performed. This gives similar feedback to that which a mobility cane might (albeit in sound), though with a longer range.

This method of increasing accessibility has enabled users of different abilities to use only the devices they need. This approach can be found in some other fast-paced accessible games such as GMA's "Shades of Doom".

## 3 Structure Adaption and Filtering

Many existing accessibility systems are tightly coupled to an underlying mainstream technology used by non-disabled people. They provide accessibility by processing the output created by a given mainstream system and adapt it for users with certain disabilities (this is the way that screenreaders and accessible PDF viewers work, for example). This can cause a number of problems such as:

- Much work is required to interpret the meaning of the original output, based on visual markup such as layout and colouring [4, 5].

– It may not be possible to directly extract or infer the types of information that specialist/disabled users require from output targeted at the "normal" user.
– When the underlying technology becomes obsolete, so does the accessibility system built on top of it[2].

The examples above fit in with many contemporary approaches, which emphasise the idea that to truly cater for the varying needs of users, the information behind any final output must be adapted for those needs. Rendering should be decoupled from information processing so that "accessibility" systems can utilise data at the same level that mainstream systems currently do.

### 3.1 Domain-Specific Solutions

In the case of the current work, this was achieved by utilising the architecture of modern computer games. Principles for *how* the information must be adapted were developed. Some domain-specific principles of structure simplification we have developed are described below. It would be very useful if the principles could be made more general; work towards achieving this is being carried out.

**Necessity-Based Rendering** — It is not required, or even useful, to bombard the user with an audio interpretation of all graphical output from the game. In the context of navigation, obstacles such as walls need only be explicitly rendered[3] when they *become obstacles* for the user. Doors or ramps, however, should always be rendered, even if they are not directly in front of the player.

**User-Centred Flexibility** — Even when considering only blind gamers, we found significant diversity in terms of preferences for rendering style. Two main schools of thought existed[4]. Within these, users had differing opinions on factors such as how quickly object indicator sounds should fall off with distance, how often scans for objects should be carried out and so on.

Note that through adhering to these principles, no information has been explicitly *added* – this does not appear to be necessary to help disabled people overcome barriers to accessibility. Our and others' work [6, 7] has shown that simply adapting the underlying information and transforming it into the most appropriate format for rendering is the most important factor in increasing accessibility.

In fact, feedback given by AudioQuake users[5] would as far as to suggest there is no such thing as a "fair advantage" – trying to overcompensate for an inaccessible system by creating supposed advantages for disabled users may simply serve to confuse them.

[2] A discussion of alternative, more loosely coupled architectures is out of the scope of this paper but is currently being investigated.
[3] This does not preclude effects such as echos from being used to give the user an impression of their immediate environment.
[4] These differences, and their implications, will be discussed in a separate paper.
[5] There is a core set of roughly 15 blind and vision-impaired gamers that take part in our electronic discussion group.

### 3.2 Case Study: The EtherScan RADAR

An example of the above phenomenon can be found in the behaviour of a navigation aid present in AudioQuake. The "ESR" warns players of nearby enemies and team mates using a RADAR-like metaphor: sounds emanate from the position of the enemy and have a gain and repetition speed proportional to the players' distance from them.

Originally it was conceived that allowing the ESR to "see" beyond walls and doors would help give blind players more chance of survival in the game. In reality, this feature – intended to help users – interfered with their sense of global navigation and caused them to track enemies behind walls, getting both stuck on obstacles and frustrated in the process.

Alternative indicators for enemies obscured by walls (such as muffled sound effects) could be used – and their effects may be investigated. However, the principle of conveying the required information in the simplest possible way and the avoidance of information overload [8] has been a prevalent (and successful) theme during the development and use of AudioQuake.

This result implies that mixing local and global navigation in an ad-hoc manner, within the same modality, can be confusing for users and detrimental to their ability in and enjoyment of games (and, thus, similar other activities).

### 3.3 Generalisation

A greater understanding of why certain pieces of information can be omitted from the auditory representation of the visual scene could be useful. This would enable us to construct a model of how people with certain disabilities (sight loss in this case) navigate, highlighting the types of information they require to do so effectively. From such a model, it would be possible to determine how this required information might be provided in this and other settings. Currently the implementation of navigation aids in AudioQuake serve as an empirical model. However, creating a more abstract and general version is an important goal and further work is being carried out in this area.

## 4 Serialisation and Prioritisation

A common generic approach to increasing the accessibility of a system is to (a) create a method for serialising its output and (b) put measures in place to ensure that the linearised output remains understandable (this may involve prioritising the rendering of parts of the output based on user needs). Examples of this are the way in which screenreaders interpret HTML pages and the work of the LAMBDA project [9]. This approach is popular and has a number of advantages, some of which are described below.

- It reduces multidimensional problems into single-dimensional problems, which may be easier to understand or at least display (the two most popular accessible output formats are speech and 1-dimensional Braille displays – both of which are linear).

- In some cases it is the more cost-effective and achievable approach, especially in areas such as web accessibility where retrofitting is more attractive to companies than redesigning their web presence[6].
- Sometimes it is the *only* known effective way of conveying information in an accessible form.

However, there are also some significant disadvantages to serialisation, which are highlighted below.

- Multiplexing nominally multidimensional data so as to render it via a 1-dimensional output medium may be cognitively demanding for users.
- Though it could be the most economically attractive approach, requiring less re-working of existing systems to implement, it may well not provide as sophisticated a level of accessibility as other methods[7].
- On its own it simply puts a (potentially large) amount of information into a format that can be "read". This does not necessarily make the information easier to understand (problems with navigation through the data may well occur). Other techniques are needed to complement serialisation to prevent information overload[8].

### 4.1 Domain-Specific Factors and Solutions

The above points mention some systems that the user can interact with largely at their own pace. However, many 3D computer games (and similar systems) pose a number of additional challenges:

- They are often fast-paced and time-critical.
- They generate a large volume of (mostly visual) information.
- This information spans multiple domains (spatial, strategy, storytelling and is presented in parallel.

AudioQuake does have some basic characteristics in common with hyper-stories such as AudioChille [11], however it is much more like other audio/accessible first-person-shooter games such as Terrafomers [2] and Demor [1], though unlike the latter it is targeted firmly at users who can only access commodity computing hardware.

[6] e.g. Screenreaders can linearise HTML tables with little effort required on behalf of the content producer. The planning agent approach [10] requires the author to undertake some extra work in order to work most autonomously (the adoption of newer mark-up standards would negate the need for a lot of this work, but that alone incurs significant effort).

[7] e.g. The planning agent approach [10] is capable of providing significantly more power and flexibility to the user than the serialisations of tables produced by a screenreader, but to work most autonomously it requires that the site be coded in XML (partly due to the semantic reinforcement afforded by meta-data).

[8] The LAMBDA project, for example, makes use of a hierarchical structure-exploration mode, for example

One major difference, however, is that it is based on technology (Quake) not originally designed for non-sighted or otherwise disabled players[9]. Though this technology's architecture provides a solid base for building accessible games, the challenges listed above are potentially much more pronounced due to the fast-paced and unforgiving nature of the gameplay. Our approach to managing these issues is centred around the following key ideas:

- We imagine that there is a certain amount of *bandwidth* available for sending data from the computer system to the user (this may vary based on user capabilities).
- Information is *streamed* from the computer to the player. Some information is more urgent than other information and must therefore be rendered in an appropriate order.
- Different *domains* of information may be sent on different "channels"; these roughly correspond to different output devices/types of output – i.e. the system is multimodal.

We present a number of complementary techniques for effective serialisation.

**Periodic Rendering by Priority** — The main tasks carried out by modern games and 3D applications run continuously. In one "tick" the screen is updated and physics rules, AI and gamecode are executed to update the user on the current state of the virtual world. Though it is appropriate to render visual, and some auditory, information at a high rate, this is not always necessary. For example, the player doesn't need to be constantly reminded of the locations of powerups (weapons, health and other such items). Indication of their position can be given periodically, as opposed to continuously.

**Sub-Domain Prioritisation** — Information in a given domain may have varying importance. For example, enemies further from the player, or out of weapon range could be rendered with a lower priority than those which are within range. Similarly for powerups.

Different schemes for prioritisation within each domain/stream were developed and found to be of use to AudioQuake players[10]

**Multimodality across Domains** — The benefits of multimodal interfaces have been discussed extensively [12, 13]. Rendering each stream to a different output device may not be possible, as there are a limited number of commodity output devices available. However, rendering all navigational information using non-speech audio and presenting communication between users using text-to-speech seems to have enabled AudioQuake players to understand the two sound-based streams separately, as intended.

The use of haptic feedback (from Braille Displays to force-feedback input devices) could further separate out streams and enable users to interact more comfortably with the system, as demonstrated by other research [14].

[9] Fortunately, it was designed well, which enables us to implement accessibility – alternative rendering – at the correct level.

[10] This should not be confused with the filtering techniques described above – even after filtering out surplus information, some prioritisation may be necessary.

**Cross-Domain Prioritisation** — When the pace of events in the game increases, the player could require more information to complete certain tasks. For example, when interacting with a number of enemies, they should ideally be informed of the position of all enemies (as opposed to just the closest) and perhaps any nearby powerups which could assist them.
This has not yet been implemented but feedback from users indicates it could be worthwhile. The general idea of load-balancing in computer networks or the grid [15] could well be applied in this situation – with the goal of sharing output bandwidth amongst modalities and devices in such a way that the user does not become overloaded with information.

More implicit, techniques for improving bandwidth usage and immersing the player more in the game are out of the scope of this paper and will be discussed separately.

## 5 Links to Education

There is a growing interest in the potential usefulness of game-like technology for education [16, 17]. This work aims to promote inclusion in a number of ways.

**Application to Other 3D Engines** — The techniques developed are quite generic and could be employed elsewhere (for games or other applications).

**Collaboration** — Though out of the scope of this paper, AudioQuake promotes accessible online collaboration, promoting integration with sighted students who may be using the mainstream version of the same technology.

**Uses in Other forms of Navigation** — Mobility training and the navigation of complex data structures are similar tasks in many respects and could make direct use of the navigation aids described here.

**Direct Educational Uses** — As well as the potential applications listed above, some more direct benefits of accessible 3D engines and games exist. They can be used as practical material in the teaching of programming, networking and even the development of important algorithms related to AI/machine-learning and searching [18].

## 6 Conclusions

We have discussed a number of issues that are key to providing accessibility to 3D environments such as computer games. Many of these challenges – relating structure/space to the user; presenting information that the user needs, quickly – are common to a number of accessibility and usability problems. We hope to generalise the currently domain-specific solutions to other areas.

## References

1. Velleman, E., van Tol, R., Huiberts, S., Verwey, H.: 3d shooting games, multimodal games, sound games and more working examples of the future of games for the blind. Lecture Notes in Computer Science **3118** (2004) 257–263

2. Westin, T.: Game accessibility case study: Terraformers - a real-time 3d graphic game. In: The Fifth International Conference on Disability, Virtual Reality and Associated Technologies. (2004)
3. Archambault, D., Burger, D.: From multimodality to multimodalities: the need for independent models. In Constantine Stephanidis, L.E.A., ed.: Proceedings of the UAHCI'01 conference "Universal Access in HCI - Towards an Informatino Society for All", New-Orleans, United States of America (2001) 227–231
4. Hanson, V.L., Richards, J.T.: A web accessibility service: update and findings. In: Assets '04: Proceedings of the 6th international ACM SIGACCESS conference on Computers and accessibility, New York, NY, USA, ACM Press (2004) 169–176
5. Pontelli, E., Xiong, W., Gupta, G., Karshmer, A.I.: A domain specific language framework for non-visual browsing of complex html structures. In: Assets '00: Proceedings of the fourth international ACM conference on Assistive technologies, New York, NY, USA, ACM Press (2000) 180–187
6. Brewster, S., Brown, L.M.: Tactons: structured tactile messages for non-visual information display. In: CRPIT '04: Proceedings of the fifth conference on Australasian user interface, Darlinghurst, Australia, Australia, Australian Computer Society, Inc. (2004) 15–23
7. Smith, A.C., Cook, J.S., Francioni, J.M., Hossain, A., Anwar, M., Rahman, M.F.: Nonvisual tool for navigating hierarchical structures. In: Assets '04: Proceedings of the 6th international ACM SIGACCESS conference on Computers and accessibility, New York, NY, USA, ACM Press (2004) 133–139
8. Brewster, S.A.: Using non-speech sound to overcome information overload. Displays **17** (1997)
9. LAMBDA Project: Linear Access to Mathematic for Braille Device and Audio-synthesis. `http://www.lambdaproject.org/` (2005)
10. Pontelli, E., Son, T.C.: Planning, reasoning, and agents for non-visual navigation of tables and frames. In: Assets '02: Proceedings of the fifth international ACM conference on Assistive technologies, New York, NY, USA, ACM Press (2002) 73–80
11. Sánchez, J., Sáenz, M.: 3d sound interactive environments for problem solving. In: Assets '05: Proceedings of the 7th international ACM SIGACCESS conference on Computers and accessibility, New York, NY, USA, ACM Press (2005) 173–179
12. Latoschik, M.E.: A user interface framework for multimodal vr interactions. In: ICMI '05: Proceedings of the 7th international conference on Multimodal interfaces, New York, NY, USA, ACM Press (2005) 76–83
13. Baljko, M.: The information-theoretic analysis of unimodal interfaces and their multimodal counterparts. In: Assets '05: Proceedings of the 7th international ACM SIGACCESS conference on Computers and accessibility, New York, NY, USA, ACM Press (2005) 28–35
14. Sjöström, C.: Using haptics in computer interfaces for blind people. In: CHI '01: CHI '01 extended abstracts on Human factors in computing systems, New York, NY, USA, ACM Press (2001) 245–246
15. Berry, R.: Performability and grid in an on demand world. In: Grid Performability Modelling and Measurement, Edinburgh, National eScience Centre (2004)
16. NESTA, EA: Futurelab. `http://www.nestafuturelab.org/research/teachingwithgames.htm` (2005)
17. Gee, J.P.: What video games have to teach us about learning and literacy. Comput. Entertain. **1**(1) (2003) 20–20
18. Laird, J.E.: Using a computer game to develop advanced ai. Computer **34** (2001) 70–75