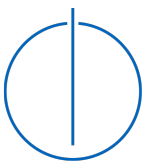
DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

Acoustic User Interfaces for Improving Situational Awareness

Monika Srdic



DEPARTMENT OF INFORMATICS

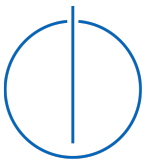
TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

Acoustic User Interfaces for Improving Situational Awareness

Akustische Benutzerschnittstellen zur Verbesserung der Situationswahrnehmung

Author:	Monika Srdic
Supervisor:	Prof. Gudrun Klinker, Ph.D.
Advisor:	Daniel Dyrda, M.Sc.
Submission Date:	March 15, 2020



I confirm that this Bachelor's Thesis in Informatics: Games Engineering is my own work and I have documented all sources and material used.

Munich, March 15, 2020 Monika Srdic

Acknowledgments

I would like to express my genuine gratitude to my advisor Daniel Dyrda for helping me find a topic for my thesis and his sincere and continuous support in the thesis process. Also I would like to thank my friends and family for supporting me emotionally for without them I wouldn't have come this far.

Abstract

This thesis presents the most important aspects and current state of user interface feedback and audio in user interfaces. Subsequent to this key factors of situational awareness in context of different user feedback forms are being elaborated. The aims of this study were to propose design approaches for auditory feedback in the context of graphical editors with focus on improving situational awareness. Additionally different interaction methods in respect to auditory feedback are discussed. The results are presented such that they form a guideline for development and design of auditive feedback in user interfaces as well as for understanding important challenges for further studies and research on this topic.

Diese Arbeit präsentiert die wichtigsten Aspekte sowie den derzeitigen Stand in dem Thema Feedback in Benutzerschnittstellen sowie Audio in Benutzeroberflächen. Außerdem werden zentrale Faktoren der Situationswahrnehmung in Bezug auf verschiedene Feedback Formen erarbeitet. Der Hauptkern der Studie besteht in der Erarbeitung und dem Design von akkustischem Feedback in Benutzerschnittstellen für gaphische Editoren mit Fokus der Verbesserung der Situationswahrnehmung. Außerdem werden verschiedene Interaktionsmethoden bezüglich des Akkustischen Feedbacks untersucht. Die Ergebnisse dieser Arbeit sind relevant für zukünftige Forschungen und Studien und stellen eine Hilfestellung und Zusammenfassung wichtigster Richtlinien und Anforderungen für Entwickler und Designer in diesem Themengebiet dar.

Contents

1 Introduction

1.1 Motivation

The improvement of human-machine-interaction is an important field of research since the usage of computers and similar technology has become essential in many areas of life. Various types of media are utilizing some form of user interface not only for aesthetics but rather for foremost communication between the user and the system. Among many advantages which are being analyzed further in the thesis user interfaces provide feedback to the user from a given system and give him an enhanced awareness of the current situation. It becomes increasingly significant to give proper feedback in respect to the given context and system state. One very common use case of human - computer interaction is the usage of a graphical editor [1, p. 1]. The user interface design and user feedback design is mostly held minimal in order to focus on the functionality of the system. Therefore multimodality is being explored little to none in context of graphical editors. Especially user feedback remains only visual although other feedback types and also combination of multiple types are already being used in many other multimedia like for example video games, apps and movies. Does acoustic feedback in context of graphical editors make sense? If so, to which extend and how can auditory feedback improve the situational awareness?

1.2 Contributions

This thesis proposes an overview of the most important principles and guidelines of user interface design, situational awareness and audio design for creating optimal auditory feedback for user interfaces of graphical editors. Additionally this work presents abstractions from most common use cases of graphical editors and analyzes them with the goal to design proper acoustic feedback for each abstraction.

1.3 Thesis Structure

The next chapter (2) covers the related work. Since there are not many references which cover the exact topic of the thesis similar use cases are being considered. Chapter

3 describes an overview of user interfaces in general and sets graphical editors into context. Chapter 4 analyzes situational awareness and presents the important key factors of it. In chapter 5 audio is being defined as well as important aspects of sound are being elaborated which then can be used for chapter 6. There four different abstractions are being made for use cases in graphical editors and for each one acoustic feedback is being designed and analyzed. The last chapter summarizes the thesis and discusses the outcome which serves as a foundation for future research.

2 Related Work

Throughout the precise research process only little to no examples were found which cover the specific research topic despite collecting information from a broad array of resources. Still similar approaches were found where acoustic feedback was designed and used for other systems than the graphical editor. Those are being illuminated here as well as put into relation to the thesis. Each example gives important insight for how acoustic feedback influences the user.

2.1 Obstacle Detection and Warning System by Acoustic Feedback for Improving Spatial Awareness

The first related work called "Assisting the Visually Impaired: Obstacle Detection and Warning System by Acoustic Feedback" introduces a multimodal way of obstacle detection for the visually impaired. An electronic travel aid with stereo vision has been created for obstacle detection using a dense disparity map combined with ground plane estimation algorithms plus filtering techniques [2, p. 1](see Figure 2.1). An audio warning is returned as acoustic feedback by the system in order to prevent the visually impaired from colliding with an obstacle. Depending on the distance of the obstacle different repetitions of one instantaneous tone are emitted by the system. The technology includes audio bone conducting which allows the user to perceive environmental sound without disturbance by acoustic feedback. The system is embraced by the users because of its high usability and the performance of the acoustic feedback. The warning system focuses primarily on the user's awareness of environmental perception in 3D space, also termed as spatial awareness. The user gets input from the environment by the warning system through its mapping of visual maps to proper audio feedback. The relation of input and feedback by the system enforces a mental model to the user which implies the higher the frequentness of the "beep sounds" [2, p. 1], the closer an object is being perceived. Such abstractions are also relevant for situational awareness since sound repetitions could be used for similar effects.

Figure 2.1: Obstacle detection system for aiding the visually impaired. [taken from 2].

2.2 Exploring the Effect and Design of Auditory Feedback in Gesture Interfaces

The second related work called "Writing with Music: Exploring the Use of Auditory Feedback in Gesture Interfaces" is researching auditory feedback in combination with pen-gesture interfaces. In an experiment the users reproduced a sequence of patterns via pen-gesture with four feedback combinations which are visual and auditory feedback combined, only visual feedback, only auditory feedback, and no feedback at all [3, p. 7]. Auditory feedback is designed based on the pen position and the direction and speed of the gesture. A specific function for calculating a complex inharmonic sound sequence for each time frame has been established in order to make the acoustic feedback distinguishable as much as possible from the environmental sound. As a result of the user study the users execute the gestures quickest and with least error rate when acoustic and visual feedback are combined. Sound increases focus and therefore productivity of the users and it is emotional appealing to them. Additionally different sound designs have been tested with the conclusion that the simpler the auditory feedback was designed, the bigger the effect turns out to be for the user. On the other hand "musical feedback is better received than one continuous tone" [3, p. 22] which implies that too simple sound designs lose the purpose and emotional appeal of sound feedback. This important insights have to be kept in mind when designing acoustic feedback in context of graphical editors as well.

2.3 Mimicking the Real World via Acoustic Feedback in User Interfaces for Higher Usability and Immersion

The last related work named "Physically Based Sound Effects for Interactive Simulation and Animation" generates realistic physically based sound effects through for complex dynamic simulations and animations. High quality continuous contact sounds have been produced using modal models driven by contact forces modeled by audio rates. Goal was to create as realistic sound effects as possible which should make the complex simulations appear more natural, realistic and qualitative through audio feedback. Additionally this work focused on automating the sound design process via creating sound algorithms with modal synthesis models, which create frequencies according to the physical phenomena like sliding, scraping, colliding and rolling. [4] As a result different models for different physical phenomena have been found and large sound modes were used to create a more convincing sound effect. This work points out the impact which good sound design has on immersion and the users emotions. Finding a good abstraction and models in sound design for specific situations results in higher immersion and realistic feeling.

3 User Interfaces

3.1 User Interfaces - The Basic Concepts

The term **user interface (UI)** comes from information technology and computer science and is composed of the two words **user** and **interface**. An interface represents the contact point between one or more users and a system where communication among them is possible. A user is in most cases a living entity that interacts with an interactive object or system which gives the user value and usefulness to some extent. A system in this context is "a group of devices or artificial objects or an organization forming a network especially for distributing something or serving a common purpose" [5]. Subsequently the user interface can be defined as "the way a computer gives information to a user or receives instructions from a user" [6]. A similar and accurate definition is the instrument "a person controls a software application or hardware device" with [7].

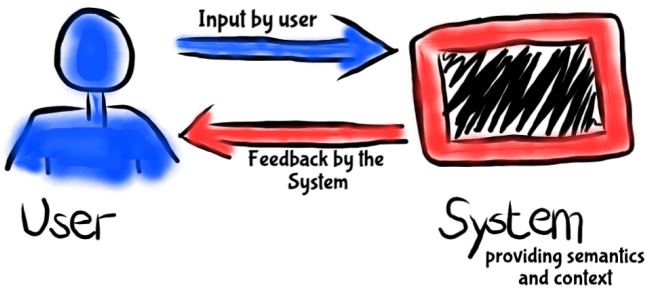


Figure 3.1: Relationship between the user and the system - User Interface Communication Flow [**Own figure 1**, **Software used: Sketch.io**].

The relationship between one and more users and the computer can be named a modular system [8]. There is input, data acquisition and management, and output. The user interacts with the PC via mouse, and keyboard while the PC provides specific semantics and context for the interaction through the UI and the display. Afterwards the PC processes the user input and generates output which has the purpose to give appropriate feedback (see Figure 3.1). Depending on the type of system and its function different user interfaces do exist. Distinctions are not only being made

between the sensory perception but also between the interaction techniques and the utility. There is a distinction between temperature and haptic, visual and acoustic UIs, as well as the distinction between graphical UI, console UI and menu driven UI. Our focus in graphical editors lies on graphical UI (GUI). Among a multitude of different input methods like controllers, buttons, mouse, camera, and many more there exist a multitude of forms of output as well. There is visual feedback, haptic feedback and acoustic feedback. The main goal of a user interface is to permanently provide the user with information about the system state and the system behaviour via giving feedback as output. Also UI gives the system a higher appeal through aesthetics as well as the ability to interact with the system.

3.2 Design Principles for Graphical UI

Many areas and tools are available for UI design. There are two important parts which need to be considered when designing graphical user interfaces. Since “vision is one of the most important senses we use and is a direct link to the world we interact with“ [9] aesthetics form a big part in the GUI design. The two big UI design tools that are used by the biggest companies in the world like Apple or Microsoft are skeuomorphism and flat design. Those two design principles are on one hand in opposition to each, but sometimes also coexist and blend. Skeuomorphism states that interface objects that mimic their real-world counterparts in how they appear and how the user can interact with them.[9] One good example for that illustrates the virtual recycle bin on any desktop. In skeuomorphism visual metaphors are used quite often to enhance engagement of the user. Skeuormophism helps to familiarize users with new technology and eases transition to new phases of increased technological consumption [9, p. 4]. Then again it also limits improvement of the GUI and the system since there is less scope for innovative designs and therefore evolvement of the system. The other design principle is flat,colourful and minimalistic. One example for that would be the Microsoft Metro UI (see Figure 3.2). While having more room for creative expression and exploration of new designs there is the danger of losing the connection to the user due to losing the realistic look and feel. A fusion of both design tool is called "Skeuominimalism" [9, p. 3] and tries to use the best of both tools as well as possible which poses a challenge.
Certainly trends to play a role in the visual design choices as well as the display resolution for quality as well as the used perspectives with which the GUI is designed. Additionally certain appeals trigger certain emotions of the user. Therefore it increases importance to choose a good GUI design tool to invoke positive emotions and to increase expressiveness of the GUI, which ultimately engages the user.

Figure 3.2: Microsoft Metro UI [taken from 10].

The second part of good GUI design focuses on the effect and impact of the user interface and especially of the output. The GUI has to be adaptive to many use cases and optimally platforms as well, flexible, consistent, clear and unambiguous in the usage and aesthetics, and invoke high user engagement. The users motivation should optimally rise so he continues to use the system because of the high quality of the GUI which enhances the user experience. One big factor is usability, the interfaces should be intuitively usable. Optimally a well designed GUI should be easy to use and enhance the learning about the system usage or the content in general. Ultimately the GUI should trigger the user to behave a certain way and provide him the information about the system state and system behaviour while remaining the correct functionality. This principle is called "principle of least surprise"[] which has the big advantage of minimizing unpredictive situations but also limits the evolvement of the design and system. A good GUI serves among being informing also as error prevention and error handling as well as aid for users in general.
Another factor which is essential for designing good and appealing GUI is that the situation of the user and the topic for which GUI is designed need to be known and specified. All of these principles aim to help understand the layers of graphical user interface design and give more insight for GUI designers.

3.3 User Interface Feedback

c. User interface feedback As already mentioned there are different forms of UI feedback. One of them is haptic feedback for example through vibration. Haptic refers to the sense of touch, anything that can be perceived through touch and tangibility. The most common feedback type is visual feedback. UI feedback is almost unimaginable without it nowadays. Visual feedback refers to anything that is perceived with the eyes. Another

feedback type which is in the rise is acoustic feedback, which is any feedback that is perceived with the ears. While it is commonly used in the games and entertainment industry, it is lacking presence in other systems for example also in graphical editors. Usually a good GUI strives for multimodality in its input and feedback. It remains important to coordinate in- and output in respect to the occurrence of interaction. Frequent interactions should reply feedback which is shorter in its duration with lower complexity whereas rare interactions should reply more memorable and unique feedback. For graphical editors the use of keyboard and mouse remains ubiquitous for triggering a certain feedback by the system.
Besides the frequency of interactions there are other properties which influence the design of GUI feedback and form feedback rules together. It is significant to balance the importance and the urgency of feedback. These factors are dependent on time and the system state. Also, the UI feedback designer has to keep in mind the relevance of the feedback. Some feedback is quite relevant, for example when giving a warning, whereas some feedback only serves as decoration and aesthetics. Feedback should be clear,adaptive and understandable, reasonable and contextual [11]. It wouldn't make sense to give something very tiny or barely perceivable as a warning feedback since the purpose gets lost. Also feedback is always structured in three parts. There is the explicit start of the feedback, the explicit end and the feedback itself with its duration. Therefore "cross-modal synchronization of timing and meaning" [12, p. 1] is essential and forms the basis of good user feedback.

3.4 Current Situation of Graphical UI and UI Feedback in Editors

Currently graphical editors remain graphical mostly since it is the main purpose of an editor. The design and development process is user centered, which means that improving the human interactions with the graphical editor is in the foreground. "Graphical User Interfaces have dominated their (of h.m. int.) interaction in most cases" [1, p. 1]. As a "powerful interactive design tool" a graphical editor reduces workload for the user which results in time saving. Feedback remains visual as well in context of graphical editors. Still visualizations can coexist around other multimedia content as well. For higher immersion and usability as well as better functionality of the system it benefits to combine multiple feedback types and to use the different senses to provide the same information to the user in an enhanced way [13].

4 Situational Awareness

4.1 Principles of Situational Awareness

Often the term **situational awareness (SA)** is associated with the military, but it is rather an omnipresent abstract term composed of the two termini **situation** and **awareness**. Situation is "a set of environmental conditions and system states with which the perception is interacting that can be characterized" [14, p. 31]. Awareness defines when a person consciously perceives information about a given situation and the person has full comprehension and understanding about the situation. Additionally in awareness the individual sentiment and past experiences play a huge role since they influence the perception of a given situation or event. Therefore situational awareness can be defined as a result of dynamic process of perceiving and comprehending events in ones environment leading to reasonable projections as to possible ways that the environment may change and permitting predictions as to what outcomes will be performing the mission [14]. Situational Awareness implies the development of a mental model of the surroundings, also called awareness model. Personality factors and expertise factors play a huge role in how the perception looks like. The mental model is formed after perceiving the complex reality. This perception needs to be understood and afterwards projected through abstraction into a conceptual map. Through communication a written or spoken report is being made which is a prediction based on the conceptual map. [14] Therefore feedback is given based on the created mental model.
Situational awareness is a fragile and difficult concept due to its level of abstraction. It takes time to be created and rebuilt and needs an organized mental model in order to have high usability. SA goes beyond immediate system awareness in its general definition. A distinction can be made between shared and individual SA. Individual SA is the SA of one individual only which is not interfering with any other mental models but is only dependent on the individual itself and the perception of the environment (see Figure 4.1). Situational Factors that influence individual SA would be for example mood, time pressure, fatigue, info and mission complexity, psychological stress, expectations and biases, misconceptions, task overload, task underload, info shortage, info overload, erroneous expectations, and lack of experience [14, p. 49]. Structural factors would be for example religion, age, sex, occupation and gender. Shared SA is a multitude of mental models which overlap and form one main SA

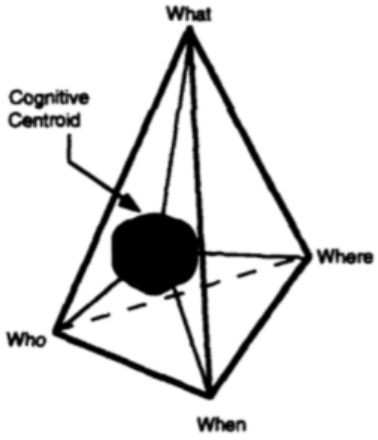


Figure 3. The "Cognitive Centroid."

Figure 4.1: Individual Situational Awareness - Concept [taken from 14].

based on group factors. Those are a false group mindset, the mentality of pressing on regardless of the circumstances, insufficient training or variable skill levels, poor personal communication skills, perception conflict, personnel turbulence, and degraded operational conditions [14, p. 49] (see Figure 4.2).
Situational Awareness in general should be clear, adaptive and accurate and mirrors reality. An advantage is that SA is very flexible and interoperable which makes it receptive to progress. On the other hand people can only consciously think about one thing at a time which makes it only scalable to a limited degree. Also there is always an uncertainty factor or factor of randomness which influences situational awareness. In addition to that there is always a set of states necessary in order for a situation to exist.

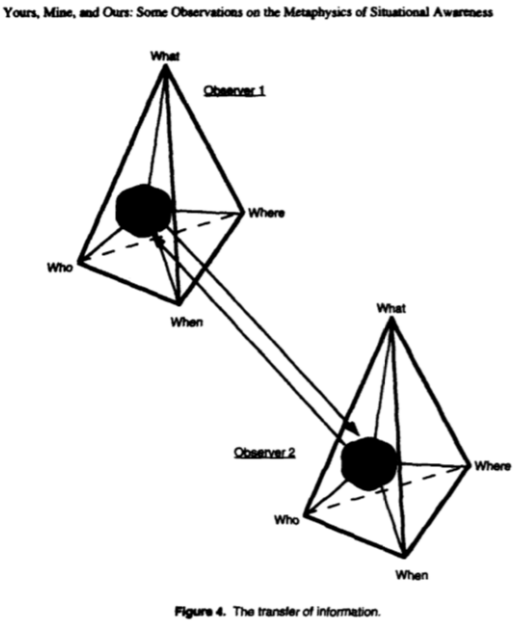


Figure 4.2: Shared Situational Awareness - Concept [taken from 14].

4.2 Situational Awareness and Spatial Awareness in Comparison

As already mentioned, situational awareness describes the process of mental model creation through a users perception which is being influenced by many factors. Often times the term situational awareness gets mixed up with the term spatial awareness. Although it remains true that both concepts focus on creating awareness about the environment, there is one significant difference. Situational Awareness on more abstract and virtual concepts like interactivity, the system state, whereas spatial awareness focuses on the physical 3D space and its properties object distance, velocity, orientation, rotation, shape size. Still both concepts can be abstracted into creating a mental model which helps the user to interact with the environment, no matter if the physical 3D space or a virtual environment like a system with states.

4.3 Situational Awareness in Graphical Editors

Since the focus lies on graphical editors situational awareness describes the awareness which the user has about the system states of the graphical editor. The system state

symbolizes the environment as well as the present situation of the system in respect to time. Since there are many specific editors nowadays some particulars or functionalities may vary but in general there are some system states which most graphical editors contain. SA in context of graphical editors describes the user's context via time, position, and task attributes. Dependent on the given context the system state can be easy or very difficult to grasp. In order to gain optimal situational awareness, it is important to identify which states and task attributes do exist in the respective situation, and which properties do the different states have which influence the user. In order to include as many common interactions in editors as possible we use the editor **Photoshop (PS)** as an example.
One common use case in graphical editors would be the process of loading something. A state can start loading, it is in the process of loading, or it ends loading either successfully or it fails. Also usually there is always selection and deselection possible. Another typical interaction with an editor would be marking something, drag and drop of something, deleting something, zooming in or out or minimizing and maximizing an object, changing layers, switching something, selecting the softness of a brush or the transparency and selecting colour. It is important to differentiate between tools and states. Not all tools are part of a distinct state, and not all states contain one tool only. Also multiple use cases may be applicable for different tools and functions of the editor.

4.4 When is Feedback Improving Situational Awareness?

Situational awareness goes hand in hand with the feedback of the system. It creates new information for the user and new input for his perception. The user gets new input from system state, new awareness is formed about the system states and the situation. The question which arises here is when is feedback appropriate and fitting for the situation? And when does good feedback add to enhanced situational awareness? - Good feedback does not only return information to the user as output but also triggers a specific behaviour and in best case also emotions in the user. Mostly those emotions are positive emotions like relief, immersion, interest, engagement and happiness. Primarily feedback should enhance usability and user experience. In connection to situational awareness this means that feedback should be on one hand appropriate and fitting the situation which the user is in at the given time, and more so it should help the user gain more clarity about the situation of the system for example about state changes or changes in behaviour of the system. When feedback adds to a adequate creation and enhancement of the users mental model of the system states, its conditions and also of the own perception of himself and what actions and behaviour is possible at this particular time then SA is improved.

4.5 Challenges Regarding Situational Awareness in Graphical Editors

Currently there is very little related work and research about situational awareness in context of graphical editors. On one hand SA gets mixed up with spatial awareness, on the other hand SA in graphical editors got lost due to many existing variations of graphical editors and many different specifications and functionalities. It remains significant to create an overview of the most important situations and use cases so that usability and user experience can be maximized, as well as the immersion. Situational awareness remains a complex hard to grasp concept, which has good chance to gain foot in a discrete environment like a graphical editor. There is much potential in enhancing the use of a graphical editor through situational awareness in combination with good feedback.

5 Audio

5.1 Defining Audio

The term **audio** describes "any sound or noise that is within a range the human ear is capable of hearing. Measured in hertz, the audio signal on a computer is generated using a sound card and is heard through speakers or headphones" [15, p. 1]. Sound and noise is the same phenomenon perceived differently. While both are single or multiple overlayed signals in form of oscillations, sound is perceived very positively whereas noise has a bad and usually repellent effect on the listener. Sound can be defined as the "acoustic consequence of kinetic events" [16, p. 2].The hearing sense is one of the most important senses that gives us feedback about the environment in order to orient oneself and get information about whether a situation is potentially dangerous. Audio perception creates awareness about the environment and therefore is highly linked with situational awareness as well as spatial awareness. Everyday life and especially in the media industry is usually hard to imagine without audio in any manner.

5.2 Properties of Sound

In the following key properties of sound are being analyzed. The atomic part of sound is a **tone** which consists in duration and pitch. In music, there is the **scale**, which is a chronological sequence of seven discrete tones. The scale forms the basis of all tones that exist. Still sound is continuous and not discrete since oscillations are continuous as well. The fundamental tone scale forms the C major scale. It consists of the tones C - D - E - F - G - A - B - C (see Figure 5.1). Since sound is continuous this scale can be arbitrarily expanded. Among the seven basic tones there are **chromatic** tones as well. A chromatic tone is the modulation of a normal tone one halftone higher or lower. From any of those a scale can be made, as well as a chromatic scale. The chromatic scale consists of the tones C - C sharp or D minor - D - D sharp or E minor - E - E sharp or F minor - G - G sharp or A minor - A - A sharp or B minor - B - B sharp or C minor and then C again (see Figure 5.2).

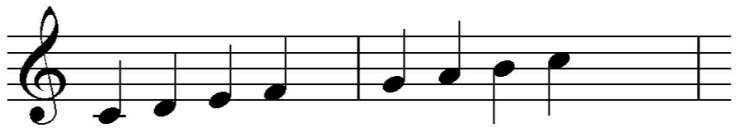


Figure 5.1: Basic Musical Scale [taken from 17].

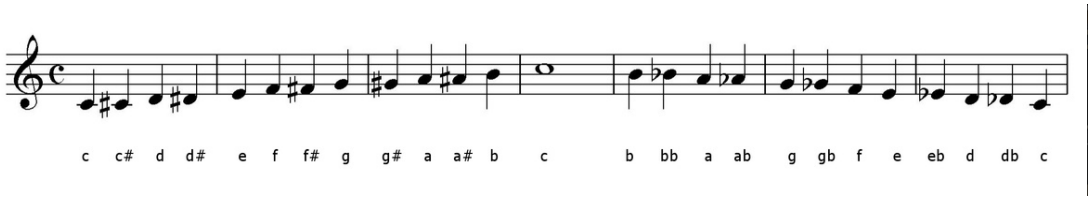


Figure 5.2: Chromatic Musical Scale [taken from 17].

A tone can have different **duration** in respect to time. In music, which is "the science or art of ordering tones or sounds in succession, in combination, and in temporal relationships to produce a composition having unity and continuity"[18],a tone duration is measured with a metronome which specifies the rhythm and speed of any audio piece. A **rhythm** "organizes time intervals by events“ [15, p. 5]. The tempo of a sound, which is measured as beats per second,goes hand in hand with rhythm and sound duration. The usual rhythm is four quarters, whereas others exist such as three quarter stroke, two quarter stroke and many more. If we consider the usual rhythm, then one tone can have the duration of all four quarters, then it it is called a whole tone, three quarters, two quarters, then it is called half tone, or one quarter only. The tone can be split up to any arbitrary length and consequently and arbitrarily stretched in duration. Playing multiple different tones at once is called playing a **chord**. In harmonics there is a specific set of tones which sound uplifting, positive and harmonic together, and an other one which sound rather melancholic and sad together. Those sets are called **major chords** and **minor chords**. They depend on the distance that the tones have to each other in the scale, also called the difference in **pitch**. This goes hand in hand with **intervals** in music and sound. The intervals are called unison, second, third, fourth, fifth, sixth, seventh and octave. In the unison the same tone is played subsequently, in

the second the distance between two tones are one whole ton, in the third the distance are two whole tones and so forth. Different intervals have different sound and different effect on the listener. While the third is usually used in the major chords and commonly triggers positive emotions, the forth usually transfers the sentiment of arrival and accomplishment. The fifth usually has a very traditional undertone and triggers a feeling of strictness and seriousness. The sixth conveys tension and stress, while the seventh transfers the sentiment of being at a break point right before closure. Higher pitch is often associated with females, something small, light, children, playfulness and positive emotions whereas lower pitch is associated with males, darkness, melancholy and sadness. In singing, there are different categories for different range of pitches. From lowest to highest, there is bass, baritone, tenor, contralto, mezzo-soprano and soprano.
A sequence of tones is called **melody**. It can be arbitrarily repeated and also varied while cycling through it multiple times. Different rhythms convey different emotions. Among the standard rhythms there are rhythmic patterns which are known from music globally for example salsa, jazz rhythms, afro beats and many more. Another property is the sound volume or loudness, also called **dynamics** of sound, which determines the intensity of the sound perception. In music there exists a basic unified set of dynamics. The term "piano" (p) gives the instruction to gently play a quiet, not very loud tone. "Pianissimo" (pp) implies to play an even more quiet tone. The more the letter p is added, the quieter and more fragile and gentle the tone has to be. Giving the instruction "forte" (f) signifies to play the tone louder, more voluminous and powerful. Here the instruction also can be arbitrarily increased. Tones can be played choppy and in staccato or coherently which would be called legato.
Multiple melodies and tones can be played at the same time by multiple instruments or sound sources, and they can be played shifted from each other, each source starting the sound at a different point in time. If the exact same sound is played that way it is called **canon**. Another property of sound is the complexity. The more complex the sound, the more variations, cycling and change in properties is made. Varying sound with pauses or complete **absence of sound** is considered as property as well. Sound pauses can be of arbitrary length. Dependent on the effect that is desired to be caused, absence of sound can create tension, surprise, silence and completion, and create an even higher dynamic as well as underline the sound that comes before or after the break if existing. Also, a distinction between **foreground and background sound** has to be made since both are perceived differently by the listener. Foreground sound is usually more in the focus of the users awareness while background sound is mostly perceived subconsciously. Combining different properties produces a unique effect of sound and may even change the initial impact of the single property.
One important aspect is also how the sound is being produced. It can be through

vocals only, **music instruments** like a guitar, a piano etc. and through **electronic sound production**. Also, environmental sound can be recorded and reused. A combination of those is possible as well. Another point of sound production is the used technology and how the sound is being played and put out. There exist different **sound systems**. One of them is stereo sound which is the commonly used sound system. The sound two or more distinct signals with two microphones can be distributed to different speakers which gives the sound a spatial effect. It gives the advantage of hearing one thing in one ear and the other sound in the other ear when wearing headphones for example It uses two audio sources. In mono sound there is only one audio signal and only one channel and the signal is put out to all speakers. It uses one audio source only (see Figure 5.2). Another sound system is surround sound which works with the 3D space and therefore more perspectives. Usually multiple speakers are set up which surround the user. Sound is measured in **hertz (Hz)** as **audio frequency**, also called audio spectrum. The human ear can perceive frequencies from 20 Hz to 20000 Hz. It also becomes important to handle **noise** or sound overload from the environment or other sources which might disturb. For this either headphones can be used or sound might have to be damped. This adds to an increased **sound quality**. The higher the quality of sound, the less noise is included.

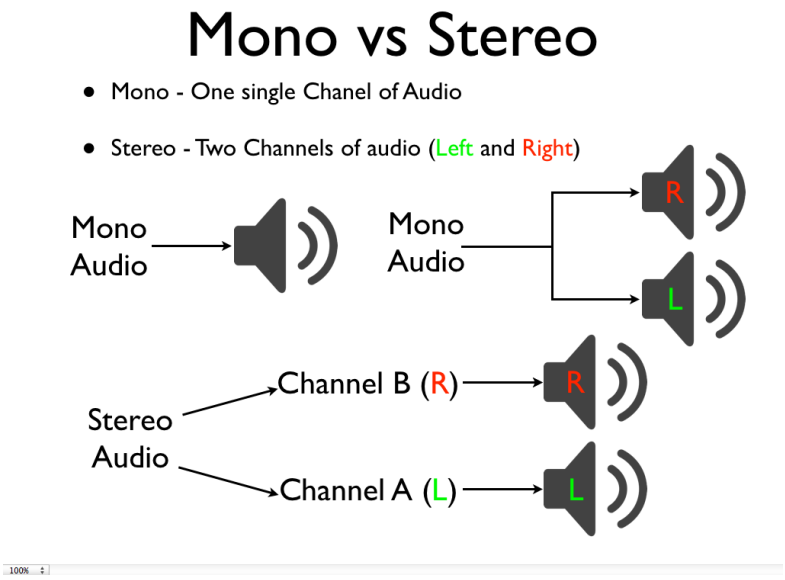


Figure 5.3: Mono Sound System and Stereo Sound System [taken from 19].

5.3 Advantages and Disadvantages of Audio

One huge advantage of audio is that it is commonly accepted and liked. "Sound provides many contextual cues that enable us to recognize important aspects of our surroundings" [20]. As huge influence in everyday life humans are used to hearing sound not only in the environment but also as part of entertainment, at work or even as a hobby with singing or playing an instrument. Music in particular has a very big appeal since it often reduces stress and brings the listener easily to a dreamy state. The immersion is very high and often people use music and sound as stimulus for getting into the flow state, which describes the state where the level of challenge and the level of the skill set is balanced. This works very well since sound in its full richness works on deep levels of subconsciousness. Therefore learning is often enhanced as well through sound since it improves effectiveness in interaction without distracting focus of attention of the human [16, p. 3]. It is a relatively simple task to produce sound in general and you get immediate feedback by hearing it. Another positive aspect is the richness of sound. Since sound is continuous there are infinite possible tones and therefore also variations of sound with its properties and levels of complexity. There is a high number of expression and design possibilities which gives room for evolving and innovation.
One the other hand most information is filtered out subconsciously and the user is not able to perceive everything of the sound with full awareness. Therefore information which is might important for the sound designer can get lost. Good sound design as well as good design of acoustic feedback is therefore essential. Also if the sound designer overdoes it or there is too much sound input for the listener, there is an overload of information and the user is overstrained by that and distracted by too much information. Also perception of sound is very individual. For some people a specific sound might be very appealing and pleasant to listen to whereas others perceive it as disturbing and repelling. It depends on the intrinsic aspects of sound which are the emotions and individual estimation which is bound to personal preferences, past experiences and much more. The degree from which on sound is perceived as poor or malfunctioning also on the individual threshold of information overload. Some people might be able to hear a specific sound for a certain duration but when this threshold is reached and the sound is played for too long then it is perceived as annoying or boring or as repelling. Also if large data of sound is being used it could be bad for the performance of the system which uses the sound.

5.4 Audio Feedback - How is it Used in Graphical Editors so far?

In graphical editors there is little to no usage of acoustic feedback so far. If so, acoustic feedback is kept very minimal and little research has been done on this particular field. The only common acoustic feedback is the "click" sound when something is selected, which usually comes from the operating system and not from the editor itself. An other example which is not related to an editor is the "bing" sound of a rendering engine when the rendering is successfully finished. In comparison to the games industry different combinations of feedback is used on a day to day basis since the user can interpret multiple info streams simultaneously. In an editor, acoustic feedback could be used to enhance interaction and immersion. Other usage of acoustic feedback in editors would be preventing the user from something, warning the user, and informing the user about the system state or behavioral change. The effects which sound has on the users emotions and psyche can be used beneficial for enhanced usability and user experience. Some information could be represented acoustically as well as defining success or failure of actions. Appropriate sound is needed for specific use cases and maybe even behavioural patterns of the user.

5.5 Key Factors in Audio Design

Sound should support the comprehension of the interface in a subtle but distinct way. It should "improve effectiveness in the interaction without distracting focus of attention of the human" [15, p. 3]. Also good audio design creates adhesion between the user and the sound. The listener can interpret multiple info streams simultaneously and has a selective hearing process which should be kept in mind as well in audio design. Most info is filtered out subconsciously through individual perception or else the user is overloaded with information. Finding the balance between fulfilling the purpose and preventing info overload is key for good audio design. It remains crucial to keep the purpose of the sound and not to overdo it and let the creative process get out of hand. Because sound perception is very individual it is a good practice to follow the common thread through answering these questions:

- What is the main purpose of the sound? Is it purely informative, warning the user or preventing the user from doing something?
- What emotions should be triggered within the user through hearing the sound?
- How frequent is the interaction that causes the acoustic feedback?

- What is the context and the situation?
- What type of sound can be used?
- What technology do I have for producing and putting out the sound?

Based on the answers to these questions it can be identified if audio feedback or audio makes sense in the specific context and how it should be designed. It would not make sense to create complex acoustic feedback for a feedback that happens quickly and very often due to information overload. For a graphical editor complex melodies or even songs are not recommended since it is a work environment and again the danger of information overload and distraction is extremely high. It stays significant to access shared mental models in context of audio and sound. For example it would not make sense to expect to trigger a positive emotion within the user with the "wilhelm scream" sound. This association with horror movies or even memes nowadays is imprinted as a mental model in our subconscious. Other examples would be loudness and very low pitched tones that imply danger, silence that implies that nothing happens and the doorbell or bell sound in general which catches the users attention because of the association with something that arrived. Audio feedback is only useful and fulfilling its purpose when it does not become noise for the user. Techniques like damping or audio bone conduction do exist which on one hand prevent important environmental sounds from getting lost, and on the other hand reduce noise in order to underline and illuminate the important audio feedback. Another solution for this problem would be the option to deactivate sound only partially or the whole sound since sound perception remains individual. Also, optimally additional measurements and design choices for people suffering from hearing impairment are being made as well, such as creating very distinct and clear sound which is loud and clear. Another distinction to be made is the sound generated by actions of the user versus the sound generated by the system. Both entities are able to produce and trigger sound as well as receive it as feedback. Ultimately the goal is the enhancement of usability and user experience which should stay priority in audio design.

6 Specification of Design Principles

6.1 When Does Audio Feedback Make Sense?

In context of graphical editors acoustic feedback only makes sense if it underlines the purpose of the feedback meaning it contributes to the comprehension and understanding of the graphical editor. Therefore the choice of whether to use audio feedback implies having a good understanding of the situation and situational awareness of the user. Many parameters do play a role in making this decision. On one hand it means understanding the states and behaviour of the graphical editor, on the other hand it means understanding the users position and his behaviour. Different factors play a role here such as the role the user plays in connection with the editor, his set of actions that can be done, as well as his emotions and his motive to use the graphical editor. As long as audio feedback adds to the synergy between the user behaviour and the utility of the editor, then audio feedback is recommended to use. Acoustic feedback therefore can help the system to trigger a certain reaction in the users behaviour such as causing new actions to happen or preventing certain erroneous actions from taking place.
Additionally audio feedback in particular makes sense when a certain emotion wants to be triggered. Since "hearing is an intimate sense similar to touch" [21, p. 3], it is important to use the right design choices for audio to trigger the right emotions. For a graphical editor such emotions would be positive emotions like relief, happiness, feeling at ease, a sense of fun and happiness about the usage, and optimally getting the user into flow state. Also in some cases audio can add to the „acoustic representation of information" [21] enhancing the visually represented information and output, and improving learning about the editor and its usage. Main goal remains impacting the designer‘s choices for the better.

6.2 Combining Audio Feedback with Graphical UI

Usually a graphical editor is mainly graphic. With audiovisual feedback another dimension is being added to the users perception. In order to beneficially use this combination of feedback "pictures and sound have to share emotional and physical characteristics" [22]. They have to share the same message and purpose, have to go

hand in hand in aesthetic, the look and feel and utility of the system. They should not stay in opposition to each other where it would only confuse the user or even worse, diminish the user experience and usability of the graphical editor and effect of the feedback.
The combination of acoustic and visual feedback gives the opportunity to explore the potential of acoustic feedback. Since the user perceives many information streams at once through his different senses, it comes natural to use different output together as well. This increases the immersion for the user since it mimics the behaviour of the everyday environment as well which also not only consists of many different impulses such as acoustic, visual, haptic and odor. Audiovisual fusing is most effective when the embodied image schemata are used by sound design that guide our perception on an unconscious level. Here, the mental model from situational awareness is relevant again, as well as the use of audiovisual metaphors. Those are usually generally part of the shared mental model of society and include symbols, feelings and objects that are associated to certain feelings, images, events or symbols. The emphasize in combining audio with graphical UI lies on reusability, compatibility and independence. Since sound effects and audio feedback are widely used and highly embraced in animation, games, and film, it gives good preconditions to assume and expect for it to happen in context of a graphical editor as well.

6.3 Elaboration of Design Principles for General Use Cases in Graphical Editors

In the following several abstractions are being described which cover most use cases of graphical editors. For each the situation is being described and according examples are being presented from existing editors like photoshop. Additionally, for each abstraction the questions from chapter 5.5. are being answered and various design choices are being presented and analyzed. It remains important to always keep the main purpose of a graphical user interface in mind as well as to use the advantages and properties of audio for optimal design choices for each abstraction. Figure 6.1 describes the abstraction of a process in a graphical editor presented in a continuous time axis. Every behaviour or action has a distinct start and a distinct end which is indicated differently dependent on the use case. Start and end can either succeed or fail. Dependent on a successful start the process and action then take place or not. Also, process and action itself can be carried out successful or not. It can be either a distinct event or a continuous action. Before and after the action there is the precondition and the post condition. Those are the system state and possible set of actions before and after the action itself. Since the focus lies on designing acoustic feedback, we assume that pre-

and post condition does not contain any sound like in the common graphical editors. This overview forms the basis for the following cases since for each the elaboration of design of acoustic feedback takes place at another point in time. Therefore each abstraction reference this graphic since different parts are being covered in each of the following abstractions (see Figure 6.7).

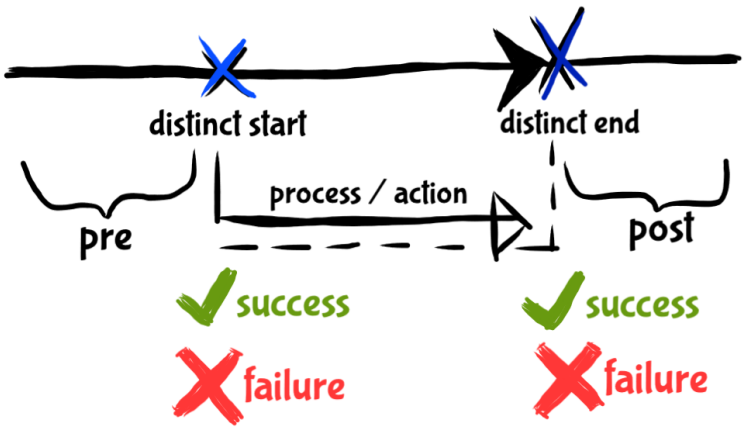


Figure 6.1: Abstraction of a Process in a Graphical Editor [**Own figure 2, Software used: Sketch.io**].

6.3.1 Parallel Usage

Parallel usage describes two processes that take place at the same time while not blocking each other from running. That usually manifest in after having started one process and at the same time being in the middle of a second process while the first one is already running. One good example for that would be on one hand using a rendering engine. While being able to continue to work on a shader or using another software, the engine renders the output. Two things that are interesting to know for the user is the progress of the rendering and also the end of the action since the visual focus lies on the other process. Feedback for the progress of the rendering and for the distinct end of the rendering would be interesting here. An other example would be saving the file. In photoshop that can take several minutes. While the user is not blocked by the action, he has the freedom to do something else. Here as well the progress and the distinct end are of interest.
In the following different sound properties are being analyzed which can be used for the acoustic feedback. First design principles are being elaborated for the **progress indication**.

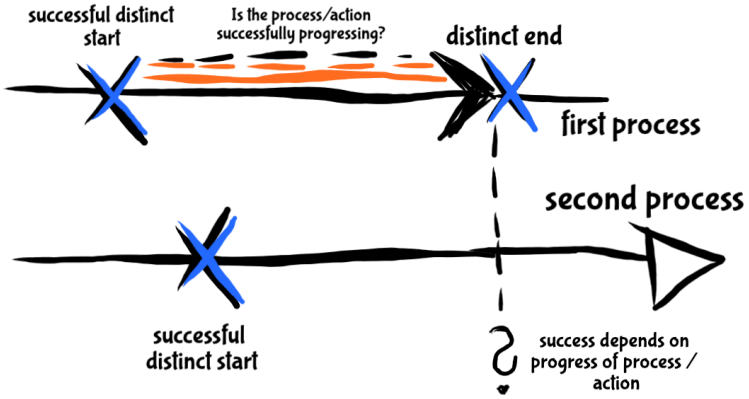


Figure 6.2: Parallel Usage - Progress Indication [**Own figure 3, Software used: Sketch.io**].

Usually if so only a loading wheel or a progress bar exists as a visual representation that something is either loading or progressing. Goal of the acoustic progress indication is to inform the user that the parallel process is still successfully running. Two parts are of significance here, on one part its the fact that the process is running, and on the other hand it is the success of the action. Both has to be kept in mind when designing acoustic feedback since this exact information and feel has to be transferred to the user. Since pre and post condition are not having any acoustic representation, and are silent, it makes sense to put out at least any form of sound to inform the user about the progress. This can be implemented in many ways. One possibility would be to **play a sound during the whole progress** to inform the user that the progress is still not finished but it is on its way successfully. Good real life examples for that would be the music in an elevator during the elevator ride or the melodies that are being played when a call is occupied and the person who is calling needs to hold the line for a while. In those two examples it has to be taken into account that the user is usually not occupied with other actions or at least not having to do other tasks which require focus and concentration. The music is trying to fill the void in these examples in order to make the waiting time less boring and to entertain the person during waiting time. For more complex situations like for example for UX designers which have to do multiple designs in one day it can be very distracting and in worst case annoying, thus minimizing the user experience and immersion, if a melody is constantly being played. Additionally it is quite a challenge to choose a good song or complex melody since the more complex a sound gets the more it is perceived differently by every individual. What one person finds very entertaining and appealing might be repelling to the other one. Another approach to using sound during the whole progress period is using something that is already known in the collective consciousness of the masses. Accessing the shared situational awareness would imply that the situational awareness of many would be accessed and users would immediately recognize that something is still in progress and not finished yet. One example for this would be the memorable theme song of the quiz show "Jeopardy!" called "Think". The problem with this and similar melodies is that they are designed for situations where the people again don not do anything else but wait. Having other complex tasks to do will most likely cause the user to get negative emotions like feeling stressed, annoyed and even angry due to information overload. Even playing the melody in a low volume would most likely miss the purpose since due to its complexity and many tones it would be perceived as noise instead of uplifting or relaxing sound.
Another design choice could be to use **only simple tones** since the less complex the more it is present in a collective mental model. Instead of playing them continuously there is a **break** after playing the simple tones in order to reduce information overload and after the break the tones are being played again. This **repetition** takes place as long

as the progress is proceeding. Here two design choices can be compared. One would be playing **one tone** and repeat it every couple of seconds. The interval would depend on how long the loading process takes. One good approach would be to use the tone G2 since it is a fifth from the basic tone C3 and due to the musical interval causing a serious and more the overall sentiment of something is happening and set in motion. But the sentiment is less associated with tension then the sixth which would be the tone A3. Other tones that are a good choice are the basic tone C3 and the tone E3. The sentiment of motion would be an audiovisual metaphor when combined with a loading wheel for example. Since the four fourth time is the most common, basic and least complex rhythm it makes sense to use this rhythm and to repeat the tones after ever four fourth. Alternatives would be every fourth, every two fourths or every three fourths. The longer the system waits until the next tone is being played, the less connected the sounds are. This can cause the effect of perceiving them as separate incoherent tones and may cause more confusion than enhanced immersion and usability. The shorter the time between the tones the more it is perceived as one whole coherent sound. Still the danger of causing information overload rises. It is essential to find a good balance. One balanced approach would be 110 bits per second as the tempo. It gives the tones enough space in between for silence but does not separate them from each other fully. Also the tones should not be held very long in order to keep it as minimal as possible. The other one would be using something **slightly more complex** but still simple enough so it is not too complex. Here again it is important to find a balance. What is too complex and what is maybe too simple and missing the purpose? The more tones are being played at the same time the more complex it is and the higher the risk of creating noise or information overload. Also playing many tones consecutively may be too distracting. One option would be playing the tones of a major chord consecutively. Recommended is here a chord consisting of only 3 tones since more only increases the potential information overload. Playing them all at once is also a possibility but will most likely be too distracting. Again, the user is doing different tasks during progress of the first process and needs as little distraction as possible. One order of tones could be G2 - C3 - E3 - the C major chord played as a sequence. Here it is a good practice to play the three tones quickly one after the other in order to have a bigger break between the tone sequences. This creates a bigger contrast and reduces information overload since the sequence is less perceived as a melody but rather as one little chord torso.
An other way to add acoustic feedback during the progress would be indicating it with voice output. Since it would create too much output if the voice would say something or sing continuously, it is recommended to play it only every quarter of the progress. When the process starts the voice could say something like "starts loading", when the progress reaches twenty five percent it could say "loaded to a quarter" or "seventy five percent left", when it reaches fifty percent it could say "half loaded" or "fifty percent

left", when it reaches seventy five percent it could say "loaded to three quarters" or "twenty five percent left". Here the type of voice plays an important role as well as the quality of the sound. Recommended is a person that speaks loud and clearly and has a soft and pleasant voice, not too high and not too low in pitch. Here as well the problem is the definition of a pleasant voice. This arbitration is perceived differently by each individual. Also, a voice can be too distracting again, since it implies the presence of another human being. Many people prefer to be alone during complex tasks and don not want to be interrupted by an other human being. Even using a robotic voice might be associated with a human too much.
All these design choices are based on the progress being successful. If it is not successful and it fails at any given point in time although it successfully started, there has to be a feedback for the user about it. Here as well something minimal is recommended. It has to be distinct from the acoustic feedback that is used for indicating successful progress. For the design choice with the chords a minor chord can be used. In order to prevent it from resulting it in creating a bad overall sentiment and melancholic feeling as effect, one can simply use a major chord but playing it as a sequence from the highest pitched tone to the lowest. In the suggested example from above that would be E3 - C3 - G2. For the voice output the solution is simple as well. The voice can say something like "loading failed". Using the design choice where only one tone is repeated with breaks in between, the solution for indicating a failed progress would be a lower pitched tone. If for example the chosen tone is G2 then the tone could be C2. It is important that the interval between the tones is again balanced, meaning it is not too high and not too low in order to make it distinguishable but not too far away from each other. This could result in confusion for the user.
From all these design choices the least complex is recommended, which is either leaving the acoustic feedback for progress indication or only using one tone which is repeated every four fourths. There should be a balance in volume as well, and it makes sense to use a very faded electronic sound for the tones. Something too impulsive or punching would be too distracting, causing the user to startle in worst case. The design choice also depends on which choice has been made for the distinct ending sound since both go hand in hand. As a sound output stereo sound is recommended since it is the commonly used technique and surround sound does not really add to the usability.

Subsequently design principles of acoustic feedback for **the distinct end of the action** are being elaborated and analyzed.

The visual representation for a successfully finished progress would be a pop up of one user interface or one icon that says "done" or "loaded" or something similar. Since this visual feedback is very short and on point, it would not make sense to go

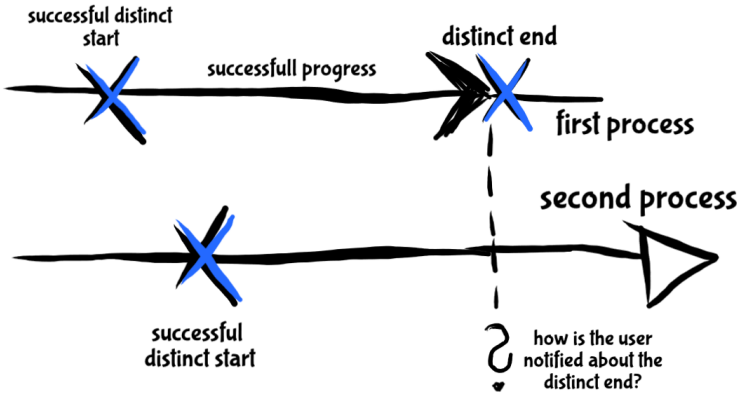


Figure 6.3: Parallel Usage - Distinct End of Action [**Own figure 4**, **Software used: Sketch.io**].

with a complex and long song or melody in this case as well. It would be too far away from the graphical representation of that information. In order to underline the visual feedback, it makes sense to use a short but distinct acoustic feedback.
Depending on which design choice has been made for the progress indication it is recommended to use a similar and matching choice for the representation of the distinct end of the process. If the not very recommended choice has been made to use a **complex melody** as progress indication, then an other melody as a feedback for the distinct end would only be too much. If the melody has not caused bad emotions and decreased immersion already, the second melody definitely will. It would have to be distinct and very different from the first one in order for the user to notice it. But even then the information load would be way too high and the melody would only distract the user from his tasks. The user will most likely be very annoyed and in worst case the sudden change of the melody would cause him to do mistakes since he is distracted. This results in highly decreased usability and a bad user experience. Also, the individual perception of the song or melody again plays a huge role. If the first melody is perceived well, the second one might not have the same effect. Then the user would even less be able to focus on the tasks since the melody is causing discomfort. Therefore if a very complex sound has been chosen, then it becomes even more important to use a very minimal and short sound as contrast. But no matter what is used it will surprise the user in a bad way since too much noise has already been perceived. If so, only a tone can be played to mark the end of the process. Best choice would be to just stop the melody. Since it has been played the user will immediately notice the absence of it. The problem that arises here is, this might work only for short

processes. If something has to load for several minutes for example, then the user has maybe already perceived the song and does not focus on it anymore, but instead he focuses on the task. Depending on how abrupt the sound stops it might not be perceived by the user since he is already used to the song and blended it out of his focus. Silencing the song would only result in the user not noticing the distinct end.
If a **chord** has been played as a sequence with pauses in between repetitively, then there are different things that can be done. One solution would be using this same chord but not playing the tones consecutively but all at once. Recommended is keeping the three tones of the chord and not extending it to a seventh chord or adding any tones to it. It will only distract the user and decrease the user experience because of information overload. The chord can be either held for a certain amount of time or played once in staccato. The second option is recommended because the more the chord is being held and the longer its duration, the more the information of this being the distinct end is getting lost, which would lead to decreased usability of the acoustic feedback. The user will not perceive this correctly and will have a distracted awareness about the situation. In order for the end to be perceived as distinct the chord has to be held a little longer than the sequence. Since the tones are played with very little time in between them to keep the pause bigger. In order for that not to be mixed up and perceived as a chord where all tones are played in the same time, the chord has to be held for at least one half quarter to make it distinct. Another solution would be adding something to the sequence. For example if the sequence G2 - C3 - E3 - pause has been played repeatedly, then the distinct end can be marked as E3 - G3 major - B. There the distinction would me more clear but the risk of information overload would be higher. It can be more annoying and distracting to the user again.
If the design choice for the progress of the process was the recommended one with repeating only **one simple tone** and a pause, then there are different design choices as well which exist as options. One would be to use again a chord. the distinct end would be clear and understandable due to the contrast between one single tone and a chord. Still, the user would most likely perceive it as distracting and annoying since it is not consistent. Where prior to the end only one tone was used, there exist three now. Due to the surprising effect the user could be easily overburdened, no matter if the chord is played as a short lived sequence or with all tones at once. An other design choice could be using only two tones and playing a quick sequence. If for example the tone G2 has been repeated, then the quick short lived sequence C3 - E3 could create good contrast but cause less information overload than using three or more tones. Another option would be using only one tone which is even better and less distracting. Using the same tone that has been repeated is not recommended since the distinction between progress and end would not be clear and could easily get lost. Playing a higher pitched tone than the repeated tone for a successful end and a low or lower minor tone for a failed

end would be a good option. For example if the tone G2 has been used, the distinct tone for the successful end could be C3 and for the failed end C2. Since one tone could get lost depending how long the progress takes, two tones are recommended. For the tone G2 for example a good acoustic feedback for a successful end would be C3 - E3 and for failed end would be E2 and C major. The halftone C major causes a feeling of discomfort which is fitting a failed situation more than a harmonious interval.
If the design choice for the progress indication was the **voice** as acoustic feedback, then the distinct successful end could be something like "successfully loaded" or "progress done". This will be easily understood, but again individual perception and taste play a huge role here as well as how distracting a voice is perceived.
Here as well stereo sound is recommended since it is the commonly used technique and surround sound does not really add to the usability.

For Parallel Input a minimal approach is recommended using stereo sound. Using a simple tone with a pause and repeating that sequence in for fourth tact in tempo 110 beats per second with a flat and not punching virtually produced sound is the best design choice among the analyzed ones. An alternative would be using silence as successful progress indication and only using failed progress and the successful end of the progress as use cases for which sound is used as acoustic feedback.

6.3.2 Input Validation

Input validation refers to an input that is being made by the user which has to be validated by the system. Therefore distinct feedback has to be put out which informs the user whether his input is valid or not. A differentiation has to be made between validation on the distinct points and validation which happens on every frame update. One example for validation on start and end would be the validation of gesture input. One common use case would be the connection of nodes. Usually when you click on one node an edge appears which then has to end in another node. This node as well has to be selected. If anything else is selected the action becomes invalid and a connection has not been created. An example for validation on every frame update would be the generation of nodes. Usually a node can only be created on a free space and can only have certain size. Here an update happens every frame about whether the users action, in this case the up- or down scaling, is still valid or not. Here feedback additionally has a preventing purpose. The user is informed about whether his action would fail if he finishes the gesture at a specific point in time. Another example for that would be the drag and drop gesture. Here as well the feedback prevents the user for example from dragging the object to an invalid position meaning somewhere where it is not possible to position the object. The goal for both validation distinctions is o

inform the user about whether the action and input is valid or not as well as if it starts end ends successfully or if it fails at a specific point in time. Differentiating between this remains important since different emotions and actions have to be triggered for each case.
In the following different sound properties are being analyzed which can be used for the acoustic feedback. First design principles are being elaborated for the **validation on distinct start/ end**.

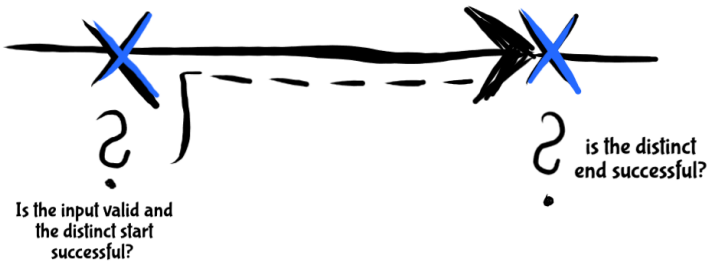


Figure 6.4: Abstraction of a Process in a Graphical Editor [**Software used: Sketch.io**].

Here, two distinct points do exist. One marks the start of the action or process and one marks the end of it. Since the act of starting and ending something is a very short and discrete event, it does not make sense to use a **complex melody** for both. That would only take away the purpose of the acoustic feedback and cause the users situational awareness to become distracted and erroneous since the user would hardly be able to map the sudden appearance of the complex sound with the fact that it belongs to the start and end of the action or process. Even if the complex melody would only last for a very short amount of time or be very short in durance or high in speed, it would rather be perceived as noise than as welcomed and increasing usability and immersion.
Another design choice would be to use **sound effects** that already exist in the collective consciousness and are known by the masses such as sound effects for movies. The problem here among the information overload is that usually sound effects are inspired by the real world and designed to be as realistic as possible. Using a realistic sound like a doorbell or similar would not fit the context of a graphical editor which is a work environment. Also it does not make any sense to use a sound effect due to the frequency of the actions. Almost any action requires input validation at least in the start, usually also at the end. Therefore input validation happens very frequently.

Hearing a widely known sound effect every time an input is validated, it would result in enormous information overload and would decrease usability rapidly. It would annoy the user to the extent where he not only does not want to produce input where it is validated, which is mostly the case, but he would most likely not want to interact with the system at all anymore which is the worst case and therefore the worst design choice together with complex melodies and songs.
Again, using simpler sound which consists up to maximum three tones per acoustic feedback results to be a better design choice here as well. Recommended is keeping consistency. If one design choice has been made for the sound of the start then similar design choices should be made for the distinct end since that increases user experience, gives the user a better chance of learning what the acoustic feedback means and it puts both points in connections and keeps the action together. Using a completely different design choice for the end for example could confuse the user easily. He could perceive it as not belonging to the action anymore but causing a different thing to happen. One design choice would be using **chords** for the distinct start and end. No matter if played as a sequence of the tones or as a chord at once, there are many variations in which the distinct start and end can be implemented acoustically. One example would be the distinct start can be a major C chord consisting of the tones C2- E2 - G2. When playing the tones consecutively from lowest to highest in the start then the distinct end could be the sequence from highest to lowest pitched tone. An example for acoustic feedback for a successful start would be C2 - E2- G2 and the successful end therefore would be G2 - E2 - C2. Using a major chord would have the purpose of marking the success of the action. In contrast and perfect consistency a failed start or end can be implemented as the according minor chord. Therefore using the major C chord as acoustic feedback for a successful start and end, we use the C minor chord as the representative for a failed start and end. The acoustic feedback for the failed start would consist of the tones C2 - E2 minor - G2 played in that order and the failed end would consist of the tones G2 - E2 minor - C2 played in that order as well. Although that design choice does not lack in consistency and as well creativity, the chance of being overburdened by that is way too high since as mentioned earlier the frequency of input validation is way too high. Using chords as a design choice here is too complex and causing confusion, stress, annoyance, information overload and therefore frustration and rejection in the user. The user will most likely be annoyed using the graphical editor after a short period of time already and in worst case never use it again which is the maximum decrease in usability.
Therefore an even simpler design choice has to be considered which is using either **one tone or silence** for representing the success or fail of the start and end of an input acoustically. For each choice the biggest point to keep in mind is again the frequency of the input validation which is extremely high. Therefore it does not make sense to

use acoustic feedback for all four use cases since the user will be annoyed and stressed very quickly by hearing the same tones over and over again. Here it is important to keep in mind the psychology of the human being. In this case it remains better to focus on what the acoustic feedback wants to trigger in the users emotions and his behaviour than what it wants to prevent. The user should feel motivated to perform much input and to interact with the graphical editor as much as possible. The user should as well have a good user experience when using the graphical editor meaning he should be immersed and have a clear understanding of the use of the editor. It should be easy to use and helping him perform the actions that he wants to take. Especially as a novice people tend to do a lot of mistakes. Later on as experts in best case users rarely do mistakes but perform nearly every action correctly. A recommended and consequent practice would be to use a **sequence of two tones**. One tone only would still catch the users attention but the meaning might get lost. With two tones the meaning of the acoustic feedback can be transferred to the user easily without creating too much sound. For every input validation on start where the start is erroneous and not valid, a quick sequence of a tone and a lower pitched half tone can be put out as feedback for example G2 and E2 minor played in this order. This halftone causes that the tone sounds inharmonic and gives the user a feeling of discomfort and stress and confusion. The user is immediately notified that something is wrong. Then as a successful and valid start it would be the best to use no acoustic feedback. It can be done the other way around as well to only use acoustic feedback for the valid start and not for the invalid start. An example would be using C2 and E2 in this order. The third is harmonic and gives a good positive feeling which optimally motivates the user and encourages him to perform more actions since the recent one was successful. Especially at the beginning the user does not want to be reminded of his mistakes every second. On the other hand usually it takes little time to learn which action is valid and which is not. Therefore using acoustic feedback for input validation at invalid start is highly recommended since it will appear less frequent in the usage of the graphical editor and when it does it is more important to be informed about that in order to learn quicker how to do it right. Same goes for the validation at the end of an action. A good example that represents acoustic feedback for a failed end would be the tones E2 minor and C2.
Another option here as well is using **voice** as acoustic feedback such as for example a voice saying "success" for successful input validation at start and similar for the other three options. The big problem here is again the huge information overload due to the high frequency of the validations.
For each design choice stereo sound is recommended here as well since it is the commonly used technique and surround sound does not increase usability and immersion. The sound is recommended to be produced virtually or electronically with a flat and not punching sound. Optimal design choice among the presented ones is choosing two

tones, one being a whole tone and the other a halftone, and playing the higher pitched one first and then the lower one. This acoustic feedback is put out only during failed input validation at start and end.

Subsequently design principles of acoustic feedback for **validation on every frame update** are being elaborated and analyzed.

Figure 6.5: Abstraction of a Process in a Graphical Editor [**Software used: Sketch.io**].

Here the frequency of the input validation plays an even bigger role since the frequency is usually much higher when validating input on every frame update. Here it remains even more important to use very **simple sounds consisting of one or maximum two distinct tones in combination with silence**. Anything else like complex songs, sounds and melodies as well as sound effects,voice or chords will result in decreased usability and user experience due to information overload.
Every frame the input can either be valid or not. There are different use cases that can appear which all will be named before describing the optimal design choice for the acoustic feedback. The input is either always valid and stays always valid until the input has been done. The input can also be invalid and stays invalid until the action has been confirmed. Other two possibilities are that it starts with being invalid or valid and changes to the respective other state via input of the user such as moving the mouse via drag and drop to a different position. For each case the design choice can be summed up into creating acoustic feedback only at the distinct points of the validation. The first time the action is validated and the input is successful then the acoustic feedback is put out. The recommended output again is the combination of two distinct tones that are both whole notes. Recommended here is the third as interval since the action is successful and valid and due to the harmonic effect of the third the user is being triggered to have positive emotions associated with harmony, success and motivation. One example for acoustic feedback would be G2- B2 played in this order. After having played it once then no acoustic feedback follows as long as the state is kept meaning

as long as the input is valid. As soon as it switches to being invalid, then an acoustic feedback takes place, which is optimally the same sequence of tones but played from higher pitched tone to the lower one in order to keep consistency but at the same time indicate that the action is failed. It would be even better to diminish the lower tone by a half tone in order to create more dissonance which creates an overall sentiment of something being off, melancholy and confusion. It instantly notifies the user that the input is invalid without scaring him. Afterwards if the input stays invalid no acoustic feedback is put out. Only when a switch happens from valid to invalid and vice versa then acoustic feedback is used. While increasing usability and underlining the important information with sound, this feedback is as free from information overload as possible.
For these design choices stereo sound is recommended here as well since it is the commonly used technique and surround sound does not increase usability and immersion.

6.3.3 Action Focus

Action focus refers to the focus that the user keeps during a specific action. In a typical graphical editor there is a pool of different actions that can be executed and usually the user does not have an overview of everything all the time. This can be on one hand restricted through the layout and design of the editor such as for example in photoshop where a user can be located in different layers during photo editing. On the other hand the user has potentially restricted his view on his own for example through zooming in a specific are, scrolling down to a specific point or similar. When performing certain actions then, it remains important to inform the user whether his actions are valid and mainly whether or not the actions affect an area or element outside of the focus view. For this feedback has to inform the user if the action was successful or not and also if possible prevent the user from doing a mistake like for example editing something out of the focus view that was valid but becomes now invalid.

In the following different design principles of acoustic feedback for **action focus** are being elaborated and analyzed on the basis of the two use cases being in different layers and changing something in the action focus area which has direct impact on something outside that area.
The possibility to be in different layers and working on them implies that only some actions are possible in one layer and some are not. Also some actions on one layer could influence previously done actions or the environment or elements from the other layers which is the same use case like the second example where no layers exist but something

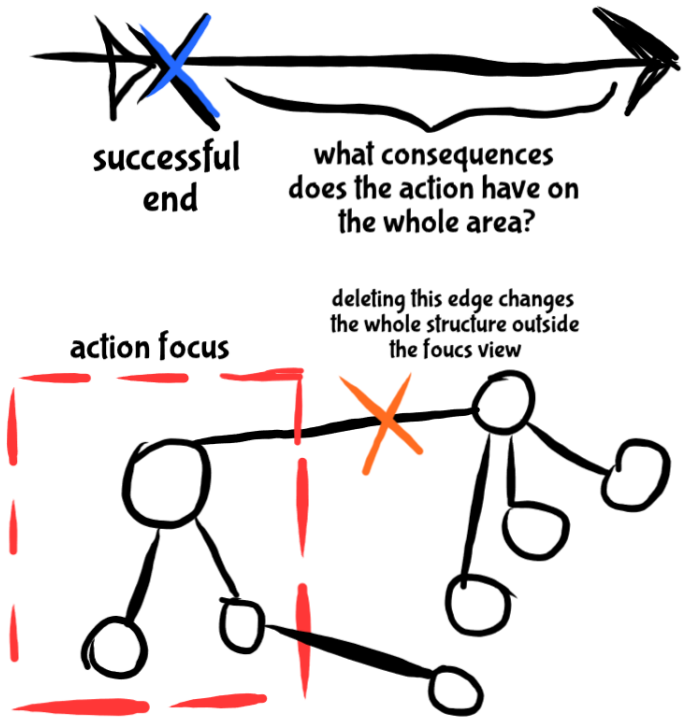


Figure 6.6: Action Focus Illustration [**Software used: Sketch.io**].

is for example zoomed in or the user focuses in one area of the editor only. We assume in the following that the full area is not visible since the graphical editor can not read minds and validate the focus of the users eyes only without eye tracking, which would be an interesting topic for future research. If an action has been taken which does not influence the outer area in any way then no acoustic feedback is required. The user only needs to be notified if something outside of his field of view changes so that he can immediately check if that was intended or not. Optimally the acoustic feedback does not play when the action has taken place but before that in order to warn the user so he can check beforehand if the impact is of meaning or not. Therefore we do not need to distinguish between successful input and invalid input but only check if the input affects any outer element before confirming the input.
Since this is not as frequent as input validation (chapter 6.3.2), but still remains in context of a graphical editor meaning it is mostly an environment where the user works and focuses on creating results with least error as possible, it is not recommended to use **complex sounds or melodies**. One good example where action focus happens frequently is the graphical editor photoshop. The user could for example focus on one little area that he zoomed in in order to mark certain spaces of the image which he wants to edit. These spaces may be part of an area outside his zoomed field. Depending on the accuracy of editing many actions can be taken that influence an outer element. Therefore using complex sound would only produce more information overload causing the usability and user experience as well as the immersion to rapidly sink.
Depending on the frequency of affecting outer elements and the outer area, voice sound and chords can be used. But only for graphical editors where the frequency of this use case is not very high since the system wants to prevent the user from making erroneous actions. As voice commands from the graphical editor, a good acoustic feedback would be something like "attention, outter area is affected" in robotic voice. The use of a human voice is also possible but more likely to be repelled since working with a graphical editor is possible for a single person only and for some people working is a private situation which should not be distracted by an unfamiliar voice. This again is very individual and difficult to grasp for the majority. Most likely it depends on the graphical editor.
Another design choice is using **simple tones**. The acoustic feedback should not distract, enhance usability and user experience but at the same time warn the user that this action will change and affect other areas. Here the shared situational awareness of the collective can be of great use. One thing that can turn out very useful and enhance usability is the repetition of a tone or little sequence for a certain amount of time. Usually this phenomenon is known from sirens of either police or the ambulance as well as from church bells that ring a certain amount of times in order to inform the

listeners about the time and also about the start of important religious fairs. One example of how it can be implemented in context of a graphical editor is using one or maximum two tones and repeating the sequence three times. If only one tone is repeated then any tone can be used, but it has to be the same one three times in order to keep the association with the mental model. A siren or an ambulance is also repeating the same tone sequence and not putting out different ones all the time. For one tone only one example for a warning sequence would look like this: A2 - A2 - A2. Even better would be the sequence with repeating two tones three times since it is even closer to the real life examples than only one tone in most cases. Here it is recommended to use the fourth as musical interval since it is usually associated with the fire department. One example for acoustic feedback with two tones played consecutively and repeated three times would be: G2 - C2 - G2 - C2 - G2 - C2. It can also start with the higher note and then be followed by the lower note and this sequence then repeated three times. If the action focus is a frequent use case then the repetition is not recommended since it would only produce more confusion and a bad sentiment in the user as well as it would overburden him. Then only one tone or maximum two tones are sufficient to inform and warn the user.
The sound technique which is recommended here is surround sound since the direction of the affected area can be illustrated through that technology as well. If the technology happens to be too expensive or disadvantageous in its installation, setup or it is simply not fitting the graphical editor then the standard stereo sound is recommended.

6.3.4 Performance Update

Another important aspect which easily gets lost and which has significance especially for developers and in the work field is the performance of a graphical editor. This has less to do with the visual aspect but still remains important since often especially for non professional users the graphical editor and therefore also the device itself can be overburdened in performance. One good example would be in coding editors or in photoshop as well. It would be useful if the user gets immediate feedback if an action or behaviour causes sudden performance drop or causes the engine or editor to transfer to an erroneous state. Typical performance properties would be the time an action takes or the amount of data which is being processed in each time step. This is helping the user to identify potential errors much quicker and giving him the opportunity to use the graphical editor more efficiently.

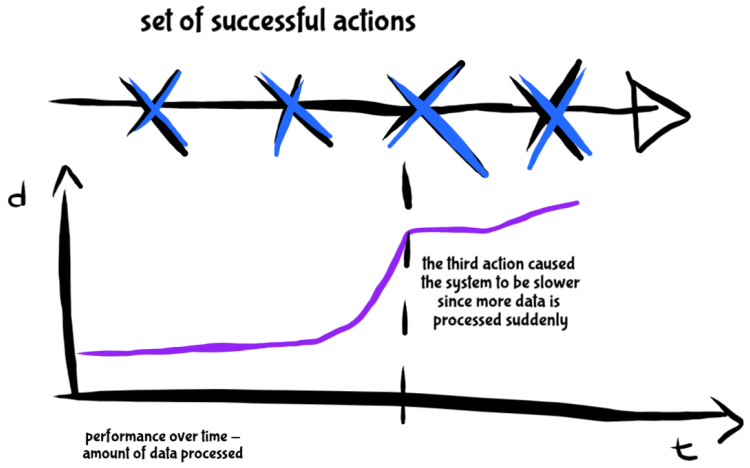


Figure 6.7: Performance Update Illustration [**Software used: Sketch.io**].

Subsequently different design principles for acoustic feedback are being elaborated for the **performance update**.
This use case is especially suited for the workplace since not every casual user is affine to technology and trying to maximize performance the same way in which it is important for companies or a developer for example. Depending on which key performance value is being measured different sound can be used. In order to reduce information and to make it more efficient there is an abstraction of things that can happen during performance measurements. Performance can drop drastically or it can increase. Therefore the performance can be in a good state or in a bad state. Of course more intermediate stages can exist such as for example performance dropping to a certain amount or at a certain point in time. Design choices are being analyzed for two examples which are the performance having two states only and the performance measuring has more than two states and it is measured with intervals. The performance can drop and if it surpasses a certain interval then an acoustic feedback is being triggered. Of course the second option is more precise and one could think about putting that more detailed information into sound as well.
For the first example it is only important if the performance suddenly becomes too bad, meaning the values surpass a certain threshold which indicates that the performance is not acceptable anymore. Complex design choices do not make sense here as well since a performance drop can happen very quickly. Therefore putting out complex sound does not only take some time which blocks the subsequent state of performance from being measured or at least put out acoustically. It also causes an additional drop in

performance most likely since complex sounds use large amount of data and take time as well. Therefore to ensure efficiency only one or two simple tones are recommended as well as well as voice sound. **Voice sound** can be such as "performance drop" or "drop", and if the performance rises again it can be such as "performance rise" or "rise". The shorter word is recommended since it saves time and resources. There is no need to put out the acoustic feedback every frame since the user only wants to know when the performance starts decreasing. It is not in the users interest to hear it every frame since it can be annoying and confusing, especially if the solution to increasing the performance is not trivial. The other design choice with using simple tones has two options. Either using one tone only per state change or two tones. One tone is shorter in time and used storage space but it can also be very inexpressive. One use case for that would be if the performance is very bad for long time and suddenly becomes better then a tone appears and the user has might forgotten the how the acoustic feedback sounded for when performance decreased making it easy to have misconceptions about the graphical editor. The user could think that the performance got lower although it got better. Therefore it is a better solution to use two tones per state change. When the performance switches from high to low then a sequence of a whole high pitch tone followed by a whole low pitch tone can illustrate that change perfectly. It is better to use whole tones and tones that are not too close in order to make the tones more distinct from each other. Additionally a too huge interval can cause confusion. Recommended intervals would be the fourth, fifth and sixth. When the performance switches from low to high then a sequence of a whole low pitch tone followed by a whole high pitch tone can represent that state change quite adequately. When there are more than two states a similar approach can be done. Either the same sounds are being used for every drop and rise, or the performance rise or drop is being represented by the amount it rise or drops. For example if the performance drops two intervals, then the acoustic feedback would be a tone and its third in lower pitch. This assumes that the musical intervals are always based on the same tone since otherwise it would cause confusion. For example if the performance drops two intervals and then later on rises three intervals then musically it could look like this: C2 - A1 - C2 - F2. The basic tone is always being played before showing the performance rise or drop.

This solution is far more creative but also more receptive to confusion and information overload. Therefore the first solution presents a better choice no matter how many states we have. If the performance drops then the a high pitched tone followed by a lower pitched tone is put out, and if the performance rises then a low pitched sound followed by a higher pitched sound is played. As a sound technique for putting out the sound stereo sound is recommended since it is the commonly used technique and surround sound is not increasing usability or immersion here. Also the sound should be produced virtually or electronically in order to keep the context of a graphical editor which operates virtually as well.

7 Conclusion

7.1 Summary

This thesis succeeded in generalizing main situations and use cases in context of a graphical editor and analyzing different design choices for acoustic feedback. While some design choices of acoustic feedback help to enhance the users situational awareness, some design choices remain purely creative and not fitting. A good guideline for graphical editors is keeping acoustic feedback less complex while balancing the frequency of the appearance of it with what it should trigger in the user. One to three tones maximum in combination with silence or pauses is enough for designing proper acoustic feedback that also enhances situational awareness. This conclusions were only able to be made after proper insight of the functionalities, key factors and principles of user interfaces in general, graphical user interfaces, situational awareness and audio with the goal to impact the designer's choices for the better and help understand the layers of user interface design. It gives a wider perspective to the topic since many aspects have to be considered in order to properly design acoustic interfaced and acoustic feedback. Graphical Editors do have grand potential to be designed in a way which improves situational awareness for the user. Many options for designing acoustic feedback exist which improve situational awareness of the user.

7.2 Outlook

Further evaluation will show which design choices for acoustic user interface feedback do improve situational awareness not only in theory but also in practice and to which extend. Among many insights it remains important to analyze as well specific editors since this approach is very general. Not all use cases are covered here, especially some edge cases since different specifications do exist. Acoustic feedback has high potential to enhance not only usability and immersion but also situational awareness. Further design choices as well as specifications of acoustic interfaces and acoustic feedback can be researched based on this thesis.

List of Figures

Bibliography

[1] G. Meixner, M. Seissler, and M. Nahler. “Udit–a graphical editor for task models”. In: Citeseer.

[2] A. Rodrıguez, J. J. Yebes, P. F. Alcantarilla, L. M. Bergasa, J. Almazán, and A. Cela. “Assisting the visually impaired: obstacle detection and warning system by acoustic feedback”. In: *Sensors* 12.12 (2012), pp. 17476–17496.

[3] T. H. Andersen and S. Zhai. ““Writing with music” Exploring the use of auditory feedback in gesture interfaces”. In: *ACM Transactions on Applied Perception (TAP)* 7.3 (2008), pp. 1–24.

[4] K. Van Den Doel, P. G. Kry, and D. K. Pai. “FoleyAutomatic: physically-based sound effects for interactive simulation and animation”. In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. 2001, pp. 537–544.

[5] *"System | Definition of System by Merriam-Webster"*. https://www.merriam-webster.com/dictionary/system. Eingesehen am 16.02.2020. URL: https://www.merriam-webster.com/dictionary/system.

[6] *"user-interface noun - Definition, pictures, pronunciation and usage notes | Oxford Advanced Learner's Dictionary at OxfordLearnersDictionaries.com"*. https://www.oxfordlearnersdictionaries.com/definition/english/user-interface. Eingesehen am 28.02.2020. URL: https://www.oxfordlearnersdictionaries.com/definition/english/user-interface.

[7] *"User Interface Definition"*. https://techterms.com/definition/user_interface. Eingesehen am 15.02.2020. URL: https://techterms.com/definition/user_interface.

[8] W. Nagel. “Content und UI Mapping”. In: *UP 2016* (2016).

[9] T. Page. “Skeuomorphism or flat design: future directions in mobile device User Interface (UI) design education.” In: *IJMLO* 8.2 (2014), pp. 130–142.

[10] *"Metro UI Bypass" Boot To Desktop Function To Arrive with Windows 8.1 | eTeknix"*. https://www.eteknix.com/metro-ui-bypass-boot-to-desktop-function-to-arrive-with-windows-8-1/. Eingesehen am 04.02.2020. URL: https://www.eteknix.com/metro-ui-bypass-boot-to-desktop-function-to-arrive-with-windows-8-1/.

[11] *"User Interface: 10 Regeln für mehr Benutzerfreundlichkeit – 247GRAD"*. https://247grad.de/blog/design-development/user-interface/. Eingesehen am 14.01.2020. URL: https://247grad.de/blog/design-development/user-interface/.

[12] D. Reitter, E. Panttaja, and F. Cummins. "UI on the fly: Generating a multimodal user interface". In: *Proceedings of HLT-NAACL 2004: Short Papers*. 2004, pp. 45–48.

[13] E. Hoggan, A. Crossan, S. A. Brewster, and T. Kaaresoja. "Audio or tactile feedback: which modality when?" In: *Proceedings of the SIGCHI conference on human factors in computing systems*. 2009, pp. 2253–2256.

[14] R. D. Gilson, L. Hitchcock, R. A. Pew, M. R. Endsley, K. Smith, R. Taylor, E. C. Adam, and N. B. Sarter. *Situational awareness in complex systems*. Tech. rep. UNIVERSITY OF CENTRAL FLORIDA ORLANDO, 1994.

[15] *"What is Audio?"* https://www.computerhope.com/jargon/a/audio.htm. Eingesehen am 17.02.2020. URL: https://www.computerhope.com/jargon/a/audio.htm.

[16] C. J. Mallett and S. Dickens. "Authenticity in formal coach education: Online postgraduate studies in sports coaching at The University of Queensland." In: *International Journal of Coaching Science* 3.2 (2009).

[17] *"Harmonielehre-Workshop 1 - Die Dur-Tonleiter :: Bonedo"*. https://www.bonedo.de/artikel/einzelansicht/harmonielehre-workshop-1-die-dur-tonleiter.html. Eingesehen am 13.02.2020. URL: https://www.bonedo.de/artikel/einzelansicht/harmonielehre-workshop-1-die-dur-tonleiter.html.

[18] *"Music | Definition of Music by Merriam-Webster"*. https://www.merriam-webster.com/dictionary/music. Eingesehen am 19.02.2020. URL: https://www.merriam-webster.com/dictionary/music.

[19] *"Audio limitations of game platforms - Sound"*. https://monkeyhammergamessound.weebly.com/audio-limitations-of-game-platforms.html. Eingesehen am 19.01.2020. URL: https://monkeyhammergamessound.weebly.com/audio-limitations-of-game-platforms.html.

[20] N. Sawhney and P. Maes. "Situational awareness from environmental sounds". In: *Project Rep. for Pattie Maes* (1997), pp. 1–7.

[21] N. Schaffert, K. Mattes, and A. O. Effenberg. "A sound design for acoustic feedback in elite sports". In: *Auditory Display*. Springer, 2009, pp. 143–165.

[22] K. Fahlenbrach. *Emotions in sound: Audiovisual metaphors in the sound design of narrative films*. 2008.