

# Introduction

*Karen Collins*

In 1983 my father brought home an IBM PC/XT. Although I recall being told that the machine was intended for work purposes (he was a mathematician), I distinctly remember having two games to play on the machine during the first few weeks the computer sat in our living room: *Frogger* (Sierra On-Line 1983) and *Floppy Frenzy* (Windmill Software 1982). *Floppy Frenzy* was a kind of educational game that was designed to teach us to keep our floppy disks away from dust and magnets, and aside from a few sound effects, I can't remember it having any music, but the *Frogger* tune (which varied on different 'platforms' – that is to say – on different computers or console machines) I can still bring to mind without any trouble at all. The simple tune that started the game, played through IBM's small speaker on one mono channel, must have been played back to me a thousand times over the following years as I, my family or my friends played the game.

Anyone who has played video games – or who has sat in a room while somebody else played a game – is familiar with the repetitive nature of games music. It is the bane of composers and players alike, who eventually resort to shutting the sound off to get a bit of peace. Games composers are well aware of the reputation of their craft as repetitive, incessant beeping, and there have been many attempts to overcome the issue as well as this reputation. Not only are games music composers finally achieving more recognition for their work, but they are increasingly interested in tackling the problems of creating sound for a participatory, dynamic media; not only creating more variability in their soundtracks, but also ensuring that the music and sound effects are responding to the needs of the game and the game player. Many of the chapters in this book touch on or discuss in detail some recent thinking and approaches to these issues.

There are, of course, reasons why games music was (and to a lesser extent continues to be) so repetitive. Early distribution systems – floppy disks, cartridges and so on – had limited amounts of space available for memory and storage (see below). Most early games were coded by hand, and writing a few simple bars would be as cumbersome as several lines of code for a single instrument voice. The competitive nature of the industry (and short shelf lives for hit titles) meant that there simply was not enough time to write lengthy songs. Moreover, most early games music composers were not musicians, but were programmers, and although many were talented, their main focus was typically not on creating great music, but on game-play and on graphics. Not least of these problems was the fact that, unlike common uses of a film or television programme, a game would typically be played over and over, for hours and hours on end, as players 'died' and restarted a level. The timings were unpredictable, since some gamers might achieve a certain degree of skill that meant longer game-play, while others were stuck on the first few minutes of

a game for hours or days. A brief history of the development of games audio and its technology will provide some context for understanding the chapters that follow.

### A very brief history of games audio<sup>1</sup>

The earliest video games – those going back to what are allegedly the first, William Higginbotham’s ‘Tennis for Two’ game of 1958, and *Spacewar!* of 1962 created at MIT – had no sound. Likewise, the earliest home consoles – such as the Magnavox Odyssey – were also silent. It was not long, however, before games were introduced into pinball arcades, and the very first mass-produced game, Nutting Associates’ *Computer Space* (1971), included a series of different ‘space battle’ sounds, including ‘rocket and thrusters engines, missiles firing, and explosions’.<sup>2</sup> Atari’s *Pong* (1972), with its beeping paddle sound, was soon to follow.

It was Taito/Midway’s *Space Invaders* of 1978 that was to introduce the idea of a continuous background soundtrack: four simple chromatic descending bass notes which repeated in a loop. Particularly notable, however, was the fact that, despite being the first to include a background track, the game’s developers also recognized the need for the soundtrack to be dynamic – to interact with the player in some way. In this case, the music’s tempo was affected by the player’s progress in the game, with the four tones increasing in speed as the game progressed.

Most video games music at the time of the early arcade hits included one- or two-channel tunes either as quick title themes or two- to three-second in-game loops, although another important invention in games audio came quickly: the ‘cut scene’, an intermission cinematic, during which the player would sit back and watch, used as a reward for attaining a particular level. Tohru Iwatani’s *Pac-Man*, the first game to incorporate a cut scene (Midway 1980), enjoyed its most musical times during these scenes, since typically the simultaneous playback of sound and player interaction was typically too taxing on the old processors, but cinematics were less demanding. *Pac-Man* also brought about a mass realization of the catchiness of sound in games – its infamous ‘waca waca’ in-game sound and opening two-channel title theme became the source of several gimmick songs in the 1980s, such as Weird Al Yankovic’s ‘Pac-Man’, and Buckner and Garcia’s ‘Pac-Man Fever’ which quickly led to an album of other games tunes, including ‘Do The Donkey Kong’, ‘Ode to a Centipede’, ‘Froggy’s Lament’ and ‘Defender’. Video games music may initially have been viewed as a gimmick, but it was soon to be taken more seriously.

Sound in games was slow to develop, however, as PCs in particular were considered business machines, and audio was not seen to have many business applications, and was therefore not a priority for computer developers. Nevertheless, there were many progressive ideas introduced to the audio programming of 8-bit machine games, and 8-bit games developed a unique aesthetic that is still enjoyed by gamers and music fans today. The majority of 8-bit machines (and early arcade

---

1 This history is an abbreviated version of that published in Karen Collins, ‘From Bits to Hits: Video Games Music Changes its Tune’, *Film International* 13 (2004): 4–19.

2 See the flyer for the game at The Arcade Flyer Archive, <http://www.arcadeflyers.com/> (accessed 10 June 2007).

and pinball machines) used sound chips known as programmable sound generators (PSGs). Early PSGs used analogue synthesis, or subtractive synthesis, which starts with a waveform created by an oscillator, and uses a filter to attenuate or subtract specific frequencies. It then passes this through an amplifier to control the envelope and amplitude of the final resulting sound. PSGs offered little control over the timbre of a sound, usually limiting sounds to single waveforms (typically square waves) without much ability to manipulate that waveform.

It was Commodore who fully recognized the importance of gaming to the home computer market, taking computers out of their formerly specialty stores and putting them into department stores and toy centres. In 1982, after the success of its VIC-20 model, Commodore released its 64K model, which would go on to become the best-selling computer of all time, selling an estimated 22 million units. The C64 was originally conceived of as a games computer, and the graphics and sound remain evidence of this. The sound chip (called SID, Sound Interface Device) was a three-voice plus noise generator chip, created by Robert Yannes, who had helped engineer the VIC-20, and would later go on to create the Digital Oscillator Chip for the Apple IIGS. Unlike other PC chips at the time, each tone on the chip could be selected from a range of waveforms – sawtooth, triangle, pulse and noise. Each tone could also be subjected to a variety of effects and programmable filters including ring modulation. *Frogger*, released in 1981 in the arcades, and 1983 for the IBM PC and C64, reveals the great differences between the machines of the times. The PC could only handle a short two-channel title tune and level intro song, with the rest of the game remaining limited to sound effects. The Commodore 64 on the other hand had a continuous background medley, including ‘Yankee Doodle’ and ‘Camp Town Races’.

Commodore had directly taken on the popular home consoles of the era, including the Mattel Intellivision, the ColecoVision and the Atari Video Computer System (VCS, later known as the 2600) which saw limited success when it was first released in 1976. In 1980, however, Atari licensed the arcade hit *Space Invaders*, which became a best seller and helped to spur on the sales of the VCS. Eventually, over 25 million VCS systems were sold, and over 120 million cartridges.<sup>3</sup> The sound chip in the VCS was known as the TIA (television interface adapter), and also handled graphics. The audio portion had two channels, with an awkward ‘polynomial counter’ to determine frequencies. A base tone was set, and then randomly divided by the system clock into 32 other note options, meaning many notes were quite out of tune, and making writing music on the Atari very difficult. Even title themes were rare, and if anything a few notes which sounded completely random might be thrown in as a title song. Many Atari VCS tunes sounded like atonal melodies strung together, with no real discernible rhythm.

Nintendo improved on the 8-bit console sound with the release of the Nintendo Entertainment System, or NES, sold in North America beginning in 1985. The NES used a built-in five-channel sound chip with one waveform for each channel – two pulse waves, a triangle wave, a noise and a sample channel. The NES came pre-packed with the game *Super Mario Bros.*, a game which would push its composer

---

3 See William Hunter, ‘The Dot Eaters Videogame History’ (2000): <http://www.emuunlim.com/doteaters/play3sta1.htm> (accessed 10 October 2004).

Koji Kondo into the spotlight in Japan. Kondo quickly mastered the limitations of the NES sound chip, managing to fill out its three tone channels with a clever use of percussion, catchy melodies and smooth looping capabilities that used slight variations to keep the song from getting monotonous as in earlier games. Most NES songs consisted of a melody, thickened out with a second channel, a bass line and percussion. Most Nintendo songs were melody-based, but some composers explored other possibilities, including Hirokazu 'Hip' Tanaka's soundtrack for the *Metroid* game, which intentionally avoided melody-based songs, instead opting for a science-fiction film music style, where sound effects and song blur together to create an eerie atmosphere.

Recognizing that gamers and musicians wanted decent quality sound from their PCs without having to go out and buy new computers, add-on third-party FM sound cards began to develop in the mid 1980s. Sound cards were designed with the gamer in mind: they generally had a joystick game port which could double as a MIDI port with an adapter. The first popular PC sound card was produced by the small Canadian company Ad Lib Multimedia in 1986. The use of FM synthesis techniques meant that game developers could now use a wider range of instruments and sounds. To boost sales, the Ad Lib card was packaged with software capable of playing back MIDI files ('Juke Box'), a MIDI sequencer program equipped with 145 preset voices ('Visual Composer') and an FM synthesis program to design sounds or instruments ('Instrument Maker').

MIDI, a protocol defined in 1983 to allow musical devices (synthesizers, keyboards, sequencers, mixing desks, computers) to be compatible in a standardized format, revolutionized the possibilities for games composing. Only commands, rather than actual sounds, are transmitted, meaning file size is very small. A MIDI command might, for instance, tell a synthesizer when to start and stop playing a note, at what volume and what pitch, and what voice, or sound, to use. The fact that MIDI was stored in code, rather than sampled sound, meant that it was a great benefit to games, since memory space was limited on the early systems. Although by the early 1990s most home computers had FM sound cards supporting MIDI, many of these sound cards were cheap, and the FM synthesis made the MIDI music sound disappointing. When CD-ROMs came out, MIDI in gaming was largely abandoned, and with it the notion of dynamic music, which would react to player and game engine input. Nevertheless, CD-ROM technology ensured that there was more room for music in games: previously most games had shipped on 3.5 inch floppy disks, and, perhaps more importantly to the game music composers, since CD-ROM audio was not reliant on a sound card's synthesis, composers could know how the music would sound on most systems, unlike with chip-based sound, which could vary greatly.

CD-ROM technology was incorporated into many of the 16-bit home consoles. The first real 16-bit console was the 1989 release by Sega, the Genesis (known in Europe as the MegaDrive). The Genesis produced many games ported from successful Sega arcade games like *Space Harrier*, *After Burner* and *Ghouls 'N Ghosts*. The system originally came packaged with the arcade hit *Altered Beast*, but soon took on Nintendo's *Mario* with their *Sonic the Hedgehog* character. The Genesis also had superior sound over the NES: it had a similar PSG chip to handle

effects and the occasional music part, as well as a Yamaha FM synthesizer chip, which had six channels of digitized stereo sound.

With the Genesis leagues ahead of the NES in capabilities, Nintendo realized that they would also have to build a 16-bit system to compete. By 1991, they had developed their Super Nintendo Entertainment System (SNES). The SNES sound module, built by Sony, consisted of several components, the most important of which was the Sony SPC-700, which acted as a co-processor with its own memory. It was essentially a wavetable synthesizer that supported eight stereo channels at programmable frequency and volume, and effects such as reverb, filters, panning and envelope generators, and with a preset stock of MIDI instruments. Wavetable synthesis uses preset digital samples of instruments, usually combined with basic analogue waveforms, creating more 'realistic' instrument sounds.

After attempts to continue working with Nintendo on a CD-ROM-based system failed, Sony went off on their own and developed their own system, the Sony PlayStation, a console which was cheap and easy to program, and therefore saw the support of more games designers. The PlayStation was enormously successful, selling over 85 million units, most likely due to its affordability and the massive library of games available. The sound chip was capable of 24 channels of 16-bit sound at CD-quality sampling rate, and allowed for real-time effects like ADSR changes, looping, reverb and pitch modulation. The PlayStation's CD-ROM drive could also play audio CDs, and in fact there were some games in which it was possible to pause the game and replace the game's audio with the player's own audio CD, a function which is now built into all Microsoft Xbox 360 consoles.

Nintendo bypassed the 32-bit machines altogether, going straight to a 64-bit release in 1996, the Nintendo 64 (N64). The N64 surpassed the PlayStation in capabilities, if not in popularity. The main processor controlled the audio, producing 16-bit stereo sound at a slightly higher sample rate than CD quality. Some games supported surround sound, and this was enhanced with the third-party add-on release of RumbleFx 3D Sound Amplifier. With the release of the 128-bit machines in the late 1990s (including the Sony PlayStation 2, the Sega DreamCast, the Nintendo Game Cube, the Xbox and so on), games were able to produce DVD quality audio, with multichannel surround sound. With large DVD-ROM games, music for a single game could fill several audio CDs, such as *Final Fantasy IX*, which was released as a four-CD set.

The latest generation of games consoles and computers (referred to typically as 'next gen' consoles) have eliminated many of the technological difficulties of earlier games. Games audio is often now recorded with full orchestras and choirs, and it has become quite common for games soundtracks to now be released as commercial music CDs or to be sold on iTunes. Although the technology is now there, the tools are not, and the more dynamic aspects of games audio (player participation, non-linearity and so on) have only begun to be explored.

Games that use music as the primary narrative component, such as *Dance Dance Revolution*, have been immensely popular, and offer many areas for interesting exploration and future development. The bizarre game *Vib Ribbon*, for instance, released in Japan in 2000 for the PlayStation, allowed the user to put in his or her own music CDs, which would then influence the generation of maps in the game.

The game scanned the user's CD and made two obstacle courses for each song (one easy and one difficult), so the game was always as varied as the music the player chose. Many other types of games are now beginning to explore the potential of including more dynamic musical elements, which is explored by many of the authors in this volume. The implications of this participatory and non-linear aspect of games audio are vast in terms of how scholars may approach games audio in the future, and the authors here hint at some of the directions that can be taken.

### The current state of games audio research

As Erica Kudisch shows in her annotated bibliography at the end of this book, despite thirty years of technological development in games, academic research into games audio has been slow to develop. There have been but a handful of academic articles published, often in conjunction with or in comparison to film music. The authors in this book come from disparate disciplines: psychology, business/industry studies, popular music(ology), computer science, communications and so on. This is, perhaps, how a new discipline should begin; with a variety of theoretical perspectives and practices that can create a diverse spectrum in which discussion and opinion can clash, develop and grow. As authors such as Ronald H. Sadoff and Rick Altman have shown in the case of film sound, at least in the early stages of research in a new area, it is necessary for academics to work with those in industry and to consult primary sources in order to understand the impact that processes and practice have on production.<sup>4</sup> With this in mind, the structure of the book that follows this Introduction is such that it interweaves articles by those in industry with research by academic scholars, many of whom are also practitioners themselves.

The book is divided into five sections. The first part, 'Industries and synergies', addresses the growing synergy, or mutually beneficial collaborative activities between video games and the popular music industry. The multibillion-dollar video games industry, while experiencing a slight fluctuation every five years or so when new consoles are typically released, has been increasing in worldwide dollar value at a time when other cultural industries, particularly the popular music industry, are decreasing. The ties between these two industries are now strengthening, as the popular music industry views games as a way to market music to its target groups, while the games industry can use popular music to appeal to specific markets, to reduce production costs and to open up new uses for music in games.

In the first chapter, 'The new MTV? Electronic Arts and "playing" music', Holly Tessler explores this relationship between the games industry and popular music, asking whether games will replace television as the primary form of media for exposure to new music. By focusing on the enthusiastic utopian vision of Electronic Arts executive Steve Schnur, Tessler explores games as a new distribution channel for popular music. In Chapter 2, 'Marketing music through computer games: the case of Poets of the Fall and *Max Payne 2*', Antti-Ville Kärjä narrows this focus with

---

4 See Rick Altman, *Silent Film Sound* (New York, 2007); Ronald H. Sadoff, 'The role of the music editor and the "temp track" as blueprint for the score, source music, and source music of films', *Popular Music* 25/2 (2006): 165–83.

a case study of the growing synergistic relationships between the music and games industries. Kärjä asks whether the success of Finnish indie rock band Poets of the Fall in 2005, despite little in the way of a marketing campaign, may be related to the use of one of their songs in the first person shooter (FPS) game, *Max Payne 2: The Fall of Max Payne* (Remedy Entertainment 2003).

Another recent platform for music in games is in the area of mobile (cellular) telephones, which is the subject of the second section of the book, 'Ringtones and mobile phones'. Although many early mobile phones contained games, these games were generally without sound, and since that time, sound has developed in fits and starts amongst a confusion of standards and platforms. Although mobile phones are, in a sense, audio devices (primarily for voice, but also increasingly as multimedia devices), sound in mobile games remains fairly primitive in comparison to their console, PC or online counterparts. Part of the reason for this slow development has been the fact that mobile games are often consumed in public, and therefore audio has played a secondary role to image in games, with phone companies such as Nokia warning developers, 'The game should be playable without the sounds. Allow silent starting of games. If intro music is implemented, there must be a way to switch it off. Prompt for sound settings at the start of the game ... Do not include loud or high-pitched sounds, have sounds on as the default, [or] use sounds that are similar to ring tones and alert tones'.<sup>5</sup>

One of the areas of music in which mobile phones have excelled has been ringtones, as phones have expanded from polyphonic chip-tunes to MP3 file formats. The ringtone industry is currently valued at about \$4.5 billion worldwide, and makes up approximately 12 per cent of all music sales. Some have touted ringtones as perhaps being able to save the music industry, and trade magazines such as *Billboard* now regularly track ringtone sales. Ringtones can cost as much as three times the price of regular music downloads, and are useful as a marketing tool for selling albums and singles. As mobile phones are increasingly capable of playing MP3 files, however, consumers are less enthusiastic about paying high prices for tracks that they may have already purchased on iTunes or another Internet music service. In Chapter 3, 'Could ringtones be more annoying?', Peter Drescher, industry composer and sound designer at Danger, Inc., first presents a speech that he gave in 2004, and then looks back at how rapidly the industry has changed, and where it may soon lead.

Lacking the tools that are available to console or PC games developers due in part to so many competing standards, music in games has remained behind in fidelity and responsiveness, particularly in terms of how the audio reacts to and interacts with the mobile game player. Working at the French National Institute for Research in Computer Science and Control (INRIA), Jacques Lemordant and Agnès Guerraz have, within their research as part of Project WAM (Web, Adaptation and Multimedia), explored tools development for mobile devices, which they present in Chapter 4, 'Indeterminate adaptive digital audio for games on mobiles'. Here they

---

5 Nokia Corporation, 'From Beeps To Soundscapes: Designing Mobile Game Audio' (2005): [http://sw.nokia.com/id/e2b3d80a-5ea7-453b-978e-1814310b4639/From\\_Beeps\\_To\\_Soundscapes\\_Designing\\_Mobile\\_Game\\_Audio\\_v1\\_0\\_en.pdf](http://sw.nokia.com/id/e2b3d80a-5ea7-453b-978e-1814310b4639/From_Beeps_To_Soundscapes_Designing_Mobile_Game_Audio_v1_0_en.pdf) (accessed 10 June 2007).

describe some of the common problems of mobile audio formats, and explore how game audio developers can build complex dynamic soundtracks for mobile games.

Though mobile phones are still struggling with conflicting standards and tools, current games consoles have to a large extent overcome many of the constraints that were experienced in previous games machines, and attention is increasingly turning towards the aesthetics and style of games audio, specifically how the music interacts with and adapts to the needs of the game-play and the game player. I have elsewhere defined this aspect of games music as dynamic, or audio that changes in relation to what happens in the game, encompassing both interactive and adaptive audio.<sup>6</sup> Interactive audio refers to sound events that occur in reaction to a player's input. In other words, the player can press a button to make the player jump, and the jumping elicits a sound. Adaptive audio, on the other hand, occurs in reaction to game-play, in terms of being set to occur in response to timings or other parameters set by the game's engine, rather than directly in response to the player. For instance, in *Super Mario Bros.* (Nintendo 1985), there is a timer that causes the music to increase in tempo when the time on a level begins to run out. The dynamic aspects of games music are an important distinguishing factor from more linear music such as that of film or television media. The third section of the book, 'Instruments and interactions', presents three chapters which address the issue of composing for dynamic media. Each of these chapters is written by authors who compose dynamic music, whether for games, Internet, or multimedia installations.

In Chapter 5, 'Theoretical approaches to composing dynamic music for video games', Danish game and Web composer Jesper Kaae provides a theoretical argument for plundering music's history, particularly the twentieth-century avant-garde, to search for approaches to variability and adaptability, which he distinguishes as two main requirements of dynamic music. Variability, or a kind of random sequencing, is used in games to increase the lifespan of a piece of music, which may need to be repeated hundreds of times for the player. By variably sequencing the music's component parts, the music becomes less repetitive. Adaptability, on the other hand, he defines as encompassing the interactive and adaptive elements of music described above.

In Chapter 6, 'Realizing groundbreaking adaptive music', Dutch sound designer Tim van Geelen provides a practical look at the problem of composing for games, giving us examples of the functions and uses of dynamic music in games and explaining some of the tools involved in creating dynamic audio in what he terms 'parallel composing', or composing songs in a linear fashion in layers that can be interchanged. Chapter 7, 'The composition-instrument: emergence, improvisation and interaction in games and new media' by interactive composer Norbert Herber, describes a composition-instrument conceptual model of a work that can play and be played simultaneously, bringing in examples from contemporary games such as Nintendo's *Electroplankton* (Nintendo 2005), in which a user can work with the game interface to create music.

---

6 Karen Collins, 'An introduction to the Participatory and Non-linear Aspects of Video Game Audio', in Stan Hawkins and John Richardson (eds), *Essays on Sound and Vision* (Helsinki, 2007).

The fourth section of the book involves new ‘Techniques and technologies’ that have arisen in the most recent generation of console releases (Xbox 360, PlayStation 3 and the Nintendo Wii). The integration of the audio into a game, particularly how games music interacts with other audio elements in a game (such as dialogue, sound effects and ambience), poses many interesting problems for games audio developers. Unlike in linear media where audio is mixed in a post-production model to a fixed image, game timings and interactions are unpredictable, and mixing must essentially be accomplished in real time. Mixing involves the interplay of all the audio components in a game to ensure that there is no overlap between frequencies, including deciding which elements should be emphasized, and which should be de-emphasized in the mix. In Chapter 8, ‘Dynamic range: subtlety and silence in video game sound’, Rob Bridgett, the sound director at Swordfish Studios, explores interactive mixing in real time, and discusses the problem of dynamic range in current games audio, suggesting some ways in which to overcome these issues.

Of course, another issue in games sound is the need for more variation in sound effects. Playing 40 hours with the same gunshot sound, or the same footsteps, can get very repetitive for the player, leading to listener fatigue, and the player’s desire to turn off sound altogether. Leonard Paul, a games and film composer working at the Vancouver Film School, addresses the issue of repetition by introducing us to granular synthesis in Chapter 9, ‘An introduction to granular synthesis in video games’. In granular synthesis, sounds can be constructed from smaller grains of sound which could be adapted in real time in a game, so that no sound effect would play back in the same way twice. He takes us through a tutorial using an open-source real-time graphical programming environment, PureData, and gives us examples of how such an application could be used in games sound effects, ambience, dialogue and music to ensure the player gets enough variation to keep sound in games interesting.

Finally, the last section of the book, ‘Audio and audience’, addresses the uses and users of games audio. The leading industry organization, the Entertainment Software Association, claims that over 75 per cent of heads of households in the United States now play computer or video games.<sup>7</sup> They also claim that games are now almost evenly divided in terms of gender (slightly more males), and that the average age of gamers is increasing as the first generation ages and buys machines that they share with their children. But audiences vary considerably depending on genre and platforms, as does the music, and how music is used in these games also varies. The final section of this book, then, asks whether or not audio has an impact on players, what this impact is and what may be some of the alternate uses for games audio.

For instance, despite some of the advances in games sound technologies, there have been some musicians who have coveted the older chip-tune aesthetic of the earlier 8-bit games era (roughly 1975–85), and who create music from older sound chip technology. These composers embrace the many constraints of the older chip technology, which was often limited to simple (typically square) waveforms, few channels and low memory. While chip-tunes, or ‘micromusic’, have been around for many years, growing out of the computer ‘demoscene’, there has been a recent

---

<sup>7</sup> See the Industry Facts at the Entertainment Software Association’s website, <http://www.theesa.com/facts/index.php> (accessed 5 June 2007).

surge in popularity for this music, particularly now, a time when there has been a general increased nostalgia for old video games (witnessed in recent art exhibits and books such as *I Am 8-Bit* and a documentary called simply '8-Bit'). In Chapter 10, 'Chip music: low tech data music sharing', chip-tunes composer Anders Carlsson of Sweden's Goto80 takes us through the history of the demoscene and chip music, and explores the implications of the recent success on the scene.

Chapter 11, Kristine Jørgensen's 'Left in the dark: playing computer games with the sound turned off', explores the impact that games sound can have on the player. Drawing on her PhD thesis work which tested male gamers responding to games with and without sound, she challenges some of the assumptions about games audio and its functions and uses in different genres, ultimately determining that game usability is affected significantly by audio, but that this is to some extent dependent on genre. In Chapter 12, 'Music theory in music games', Peter Shultz asks what kinds of skills and concepts rhythm-action games (music-based games) entail. He explores the systems of notation in music games, thus exploring games music as a potential avenue for education.