Ombretta notes

MIDI - notes from video lesson

- Used only for western music
- Used to play more than one instruments and to play semitones (it cannot represent quarters of tones)
- It represents scores and using only text messages = really easy to compress this data and to use it
    - Audio requires a lot of bits
    - But MIDI does not, since it cannot represent voice or noise
- It requires a synthesizer representing the music and the notes
- Basically like a pianist playing the piano or a guitarist playing guitar
- Used for music for the Web, given it requires very few space or for background music (e.g. supermarket/airport/videogames music)
- Quality is not so high with this scripting language and uses events (notes) with a local synthesizer
- Quality of the psychoacoustic model = number of tracks it contains. The tracks perceived by the user depends on where the file is played (so to determine high/low quality)
- MIDI is very complex: keyboard controller, sequencer (synchronizing all of the music events) and the drum machine (created for piano bar); since it does not play well for trumpets, the drum machine was a tentative to improve the quality played by the synthesizer
- MIDI sequencer
    - Responsible to synchronize musical events: to reproduce and to record music used by a performer
    - Used a lot in music composition
    - Allows to use easily other parameters
    - Very easy to simply change the message to change the instrument for a song
- Used a lot today for composing music
- Create the song and ask the computer to perform it
- It's only the synthesizer influencing the quality of the sound (synthesizer or synthesis devices)
- MIDI is organized between channels, tracks and pitches.

Channels = means through which send music events (usually a score with instruments playing). Big limit is that it uses 60 channels (= 60 instruments in the same moment at most - good for

pop, not so good for classical music)

Tracks = sequence of MIDI messages (used for score) = songs to be played with right hand and one for the left hand. Each track is associated to one or more channels (the same track can be played to more instruments at the same time)

Each channel is associated to one patch, so it is the timbre produced by the generator. A patch is a set of 128 pitches and each patch is the sample of a particular note played by a particular instruments. The piano has 88 keys but the limitation is the number of channels

Each track is associated to more channels and each channel is associated to one patch. If I want to change, I will simply send a message to see where to send stuff

How many types of MIDI messages?

- Channel messages = create the score (voice messages = note start/note end/effect for note = vibrato effect and other info about notes; mode messages are info more general about the system)
- System messages = not for a particular channel, but the whole system answers to this message, e.g. change the rhythm. The whole system has to answer to this message, since all the channels and all the patches must decrease/increase the rhythm accordingly

How the messages are organized?

- Each message requires at least 1B (usually 2B), with one byte describing the message and the following byte is the payload
- Usually, each message requires 1b (between each byte there is one bit for synchronization)

1 bit
8 bits for the type of the message
1 bit

Each message requires 10b at least; 1 bit = message is starting, 10 bit = note plays, 1 bit at the end = divide the message

10 bit and then 10 bit for the following byte = 20 b for each message (8 bits to understand which is the message and 1 bit at the beginning/1 bit at the end to divide the message)

Channel message = without end/start bit, message is divided into two nibbles (always all 1 if it is a system message, channel message has a starting 1 and then other bits)
You can understand from the first byte the type of message considering the previous thing said. This is the reason why we have 16 channels; only 4 bits to discriminate between them and only 8 possible channel messages.

Channel message = number from 1 to 8 and only 8 possible channel messages (3b)
System message = 9 and second nibble contains channel o which is directed and only 16 possible system messages (4b)

128 patches = 7 bits for the index

Mode messages which ask to the system to change the mode to which the system answers to the message arriving

Omni on/omni off = all the channels answer to all the messages (if I send this note, but is played by all patches - omni off = only the channel associated will play that note)

Poly = all channels
Mono = one channel
Both are sent to test the system

Omni off - Poly = Only the channel plays the track associated, with polyphonic
General MIDI Mode = a lot of patches but no standard way to associate patches to a particular channel

This is used for example when writing a song for the guitar but then it is played for the piano. The general mode is used to be sure that the MIDI file uses exactly the same instrument in all the systems

System messages = Divided into common messages (general functions/positioning/track selection), real-time messages (synchronization/rhythm of playback) and exclusive (not standard but add features for particular manufacturers

Why and when?

- Efficient way to encode musical sound without encoding music, since it encodes score (thanks to the synthesizer); if we are interested in soundtracks or whatever, MIDI is a good solution
- Encodes only standard music, not sounds like voice/noise/phenomena
- Quality depends on the synthesizer

Script

Notes:

- Non-musical apps for MIDI: The MIDI Protocol - Musical Instrument Digital Interface | PPT (slideshare.net)