

# Chapter 1

## Evaluation

This chapter shall discuss the effectiveness of the Excello implementation with regards to the success criteria and by showcasing examples. The conversion of an existing MIDI corpus to the Excello notation using the converter from MIDI will demonstrate the expressiveness of the notation. Next I shall explain the summative evaluation and use the data from this to assess the features implemented in the participatory design process and to reason about Excello using the CDN framework [4]. Finally, the ethics and data handling procedures shall be covered.

### 1.1 Excello Success

Both a notation for music expression and a program integrated into Excel for the playback of this expression has been implemented. As required by the success criteria, users can play multiple notes and chords of different durations. These can be combined into looped sequences with a defined tempo. In the participatory design process additional features were added as extensions. Defining multiple successive notes in a cell, turtles calculating how far they should move and nested instructions with repeats are additional features facilitating more efficient notation. Custom Excel functions, a chord adding tool and faster turtle toggling allows users to work more efficiently. Figure 1.1 shows Excello in use with participant's arrangement.

The first section of Piano Phase by Steve Reich consists of two equal piano melodies, one slightly faster than the other. The two parts move out of phase before aligning at different offsets. This is included as an example for many end-user programming tools. This is implemented in Manhattan using three rows of 24 columns [6]. Sonic Pi defined the notes in one line and eight additional lines are required for playback. Piano Phase can not be concisely notated by western staff notation. Excello only requires two cells to define two turtles of different speeds in addition to the notes. All three implementations are shown in figure 1.2.

The screenshot displays the Excello application window. The main area is a spreadsheet with columns labeled A through J and rows numbered 1 through 33. The spreadsheet is organized into sections: 'Don't you worry 'bout a thing, Stevie Wonder, arr. Redacted' (rows 1-2), 'Chords' (rows 3-11), 'Bass' (rows 12-15), 'Click' (rows 16-17), 'Melody' (rows 18-22), and 'Refrain' (rows 23-33). Each section contains musical notation and chord symbols. To the right of the spreadsheet is a control panel titled 'Excello'. It includes a 'Select Sheet' dropdown menu set to 'Don't you worry', a 'refresh' button, and a 'Turtles' section with 'Play', 'Stop', and 'Toggle activation' buttons. Below these are 'Live Turtles' listed as C3, C12, C16, and C19. The 'Insert Chords' section has dropdowns for 'Note' (set to C), 'Type' (set to 7), 'Inversion' (set to root), and 'Octave' (set to 4), along with an 'Insert' button. At the bottom of the window, a track bar shows 'Christmas.mid' with a 'Christmas' button highlighted, and other buttons for 'Mozart', 'What about me', 'Lingus', and 'Don't you worry'.

Figure 1.1: An arrangement with separated and labelled parts per instrument. Turtles refer to a global tempo at the top of the spreadsheet.

## 1.2 MIDI Corpus Conversion

Whilst being able concisely notate music western notation and other end-user programming systems do not facilitate, Excello can exactly express any combination of concurrent note onset and offsets. Therefore any piece defined in MIDI can be expressed in Excello. In MIDI allows tempo can be redefined within a track, this is not be accounted for. If instead the time between note onsets and offsets is adjusted, the uncompressed file will account for this but compressing algorithm will fail as the difference between notes may become a non-integer multiple of the minimum. Instrument specific effects such as piano pedals are not supported. This naive conversion can result in unwieldy spreadsheet sizes. One conversion method compresses the representation by dividing the difference between notes by the minimum difference in not onset between any two notes. Provided the difference between any two notes is a multiple of the minimum difference, this compression method is lossless, whilst resulting in spreadsheets using orders of magnitude fewer cells. Therefore this method would not accurately convert quavers against triplets (three notes played in the same time as two) provided these notes were not multiples of a smaller note. Given the lengths of MIDI notes can be different to the time the note occupies in notation, to automate the compression, an assumption on the ratio of note lengths was required. The modal compression algorithm is lossy if the minimum note distance is not the modal distance. This is useful if there are ornaments or note within a piece that dramatically decrease the minimum distance but occur infrequently. Therefore their loss may be tolerable for a more efficient representation.

	01:Global	02:Piano 1	03:Piano 2
000	Phase 1.0 S80	Notes	
001	...	E-5 Pn 40 .00	C#6 Pn 40 SD1
002	...	F#5 Pn 40 .00	B-5 Pn 40 SD1
003	...	C#6 Pn 40 .00	F#5 Pn 40 SD1
004	...	B-5 Pn 40 .00	D-6 Pn 40 SD1
005	...	F#5 Pn 40 .00	C#6 Pn 40 SD1
006	...	D-6 Pn 40 .00	E-5 Pn 40 SD1
007	...	E-5 Pn 40 .00	F#5 Pn 40 SD1
008	...	F#5 Pn 40 .00	B-5 Pn 40 SD1
009	...	C#6 Pn 40 .00	D-6 Pn 40 SD1
010	...	B-5 Pn 40 .00	E-5 Pn 40 SD1
011	...	E-5 Pn 40 .00	F#5 Pn 40 SD1
012	...	F#5 Pn 40 .00	B-5 Pn 40 SD1
013	...	C#6 Pn 40 .00	D-6 Pn 40 SD1
014	...	B-5 Pn 40 .00	E-5 Pn 40 SD1
015	...	E-5 Pn 40 .00	F#5 Pn 40 SD1
016	...	F#5 Pn 40 .00	B-5 Pn 40 SD1
017	...	C#6 Pn 40 .00	D-6 Pn 40 SD1
018	...	B-5 Pn 40 .00	E-5 Pn 40 SD1
019	...	E-5 Pn 40 .00	F#5 Pn 40 SD1
020	...	F#5 Pn 40 .00	B-5 Pn 40 SD1
021	...	C#6 Pn 40 .00	D-6 Pn 40 SD1
022	...	B-5 Pn 40 .00	E-5 Pn 40 SD1
023	...	E-5 Pn 40 .00	F#5 Pn 40 SD1

- (a) Column 01 keeps track of the phase, 02 defines the notes and 03 is the phased notes - defined with formulae that update depending on the phase and defined notes.

```

1 notes = (ring :E4, :Fs4, :B4, :Cs5, :D5,
2 | | | :Fs4, :E4, :Cs5, :B4, :Fs4, :D5, :Cs5)
3
4 live_loop :slow do
5   play notes.tick, release: 0.1
6   sleep 0.2
7 end
8
9 live_loop :faster do
10  play notes.tick, release: 0.1
11  sleep 0.195
12 end

```

- (b) The defined notes are played by two concurrent loops with different gaps between each note.

	A	B	C	D	E	F	G	H	I	J	K	L
1	lturtle(a3, r m*, 320)											
2	lturtle(a3, r m*, 315)											
3	E4	F#	B	C#5	D	F#4	E	C#5	B4	F#	D5	C#

- (c) Two turtles play the same notes at different speeds.

Figure 1.2: Implementations of Steve Reich's Piano Phase in a) Manhattan, b) Sonic Pi, c) Excello

I have converted three MIDI corpora. The first is a collection of 497 Bach chorales<sup>1</sup> made by Margaret Greentree. The second is 277 piano pieces<sup>2</sup> help by Bernd Krueger under a creative commons license. Finally 194 Bach pieces made available from "A Johann Sebastian Bach Midi Page"<sup>3</sup>. This is not all the files available from this site as some were not readable by the python MIDI reader. All 968 MIDI files were converted using all three methods.

The language of Excello is expressive enough to represent MIDI files and can do so concisely provided the condition of minimum note onset differences is maintained.

## 1.3 Summative Evaluation Sessions

Of the 21 users who participated in formative evaluation, 19 had continued using Excello. These users all filled out a summative evaluation questionnaire. First a review of the features that had been added since the initial sessions was given. To ensure users had a sufficient understanding of the interface before giving feedback, a short transcription task involving some original composition by the users was given.

The questionnaire first assessed the success of the features added during the participatory design process by comparing the interface before and after a feature had been added. Seven-point Likert scale questions were given to test if the issues had been solved and if overall the change rendered the system more preferable. The remaining questions were

<sup>1</sup> Accessed from <https://github.com/jamesrobertlloyd/infinite-bach/tree/master/data/chorales/midi>

<sup>2</sup> <http://piano-midi.de/midis.htm>






<sup>3</sup> <http://www.bachcentral.com/midiindex.html>

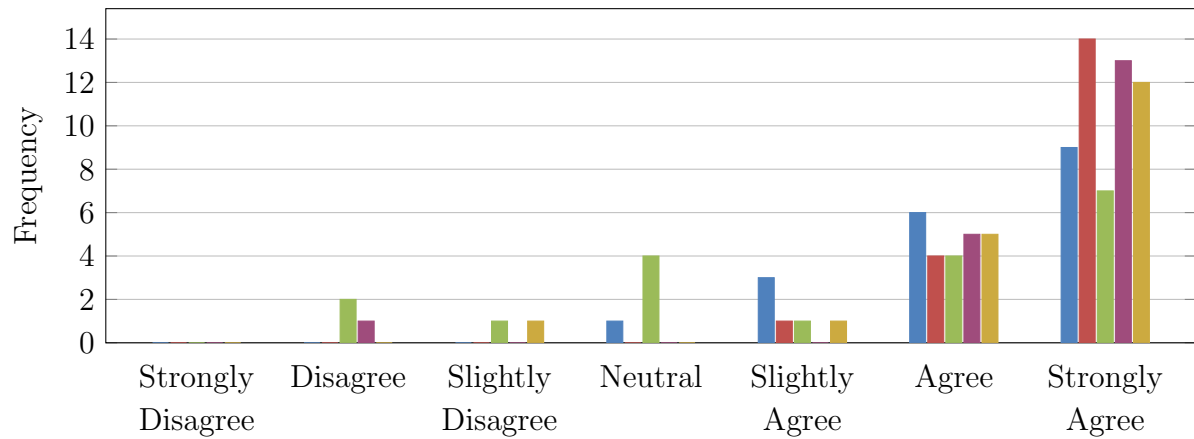
based on Blackwell and Green’s CDN questionnaire [2]. CDNs can be used to analyse musical notation [3] in addition to software systems [5], therefore it is a suitable tool for the discussion of the Excelllo notation and interface. Because dimensions have different significances for the different activities [4], users identified what percentage of their time they spent carrying out the five different user activities (searching for information, translating, incrementation, modification and exploratory design). Likert scale questions focusing on closeness of mapping, consistency, secondary notation, viscosity and visibility were used as planned in the project proposal. It was suspected that reasoning about cognitive dimensions would be more challenging for participants, so to reduce the expected variance, only a five-point Lichard scale was used. Usage and CDN questions were also answered with respect to the user’s preferred interface for music manipulation, composition or transcription. 12 users chose Sibelius, which shall be used for comparison.

## 1.4 Success of Participatory Design

For each feature added, Excelllo with this feature (system 2) was compared to first prototype implementation. The following charts show the frequency of Likart scale responses for each question. I considered the mode of the Likert scale [1]. A Chi-Squared goodness-of-fit test tests if the distribution is significantly different to uniform. As all expected values must be greater than 1 and 80% greater than or equal to five [7] and the expected frequency for one result is  $19/7 \approx 2.7$ , I combine Strongly Disagree with Disagree, Strong Agree with Agree and the remaining three options into a third group. The p-value given is from the Chi-Squared test using these three categories.

### 1.4.1 Dynamics in the Cell

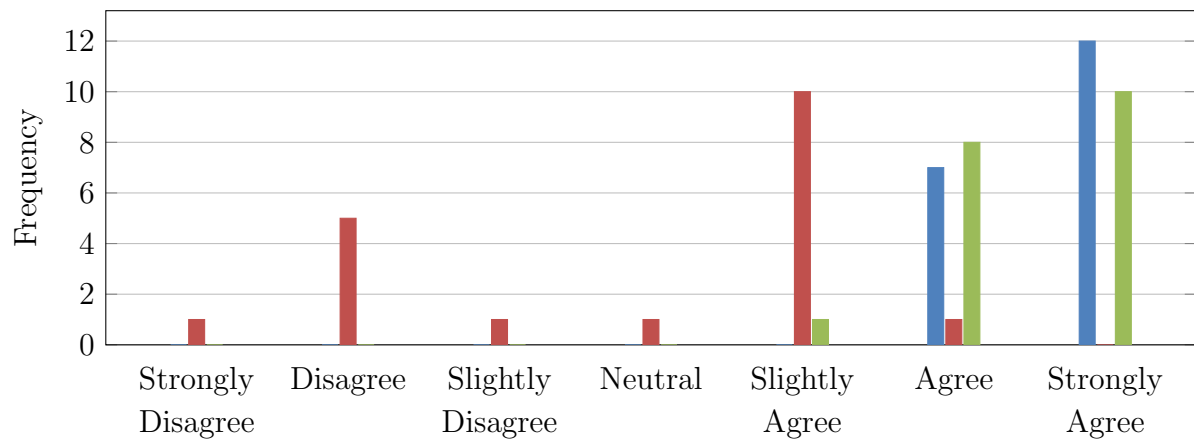
Statement	Mode	p-value
 It is easier to figure out the turtles path	Strongly Agree	0.0000
 It is easier to figure out what dynamics different notes are played at	Strongly Agree	0.0146
 It is easier to tell the order in which dynamics are applied	Strongly Agree	0.0000
 It is easier to write dynamics in the correct place	Strongly Agree	0.0000
 Overall system 2 is preferable	Strongly Agree	0.0000



The features implemented in order to solve the issues identified during participatory design, resulting in a significantly improved system users strongly agreed was improved.

### 1.4.2 Inferred Octave

Statement	Mode	p-value
less effort is required to write a part	Strongly Agree	0.0000
it is harder to figure out what octave a note will be played in	Slightly Agree	0.0639
Overall, system 2 is preferable	Strongly Agree	0.0000

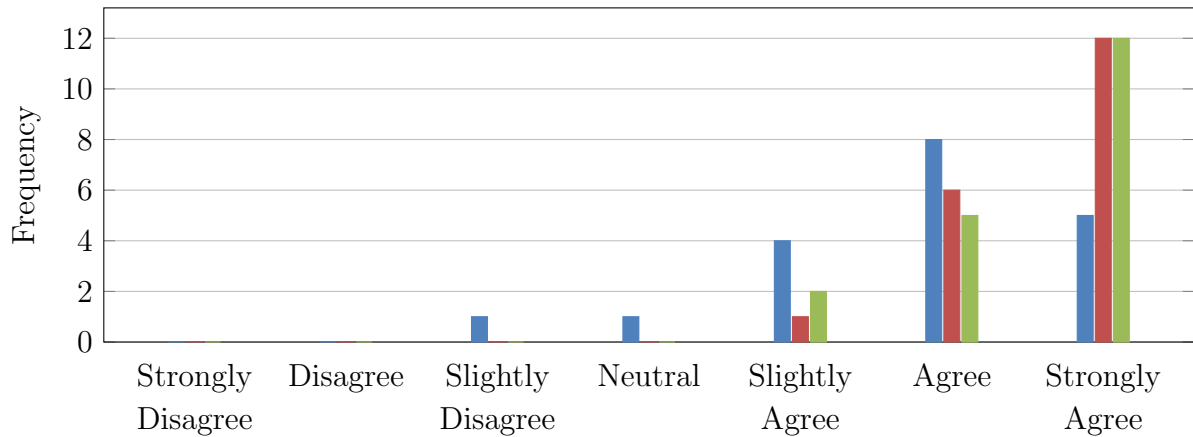


Depending on its use, the inferred octave notation has the tradeoff that the octave is less easier inferred. There were no strong agreements and only one agreement that this was the case but the modal response was slight agreement. However many users didn't find this an issue and the distribution of responses is not significantly different to uniform. Overall this addition offered a significant improvement.

Statement	Mode	p-value
■ it is easier to parse the turtle instruction and tell what it will do.	Agree	0.0003
■ it is easier to repeat sections of notes.	Strongly Agree	0.0000
■ Overall, system 2 is preferable.	Strongly Agree	0.0000

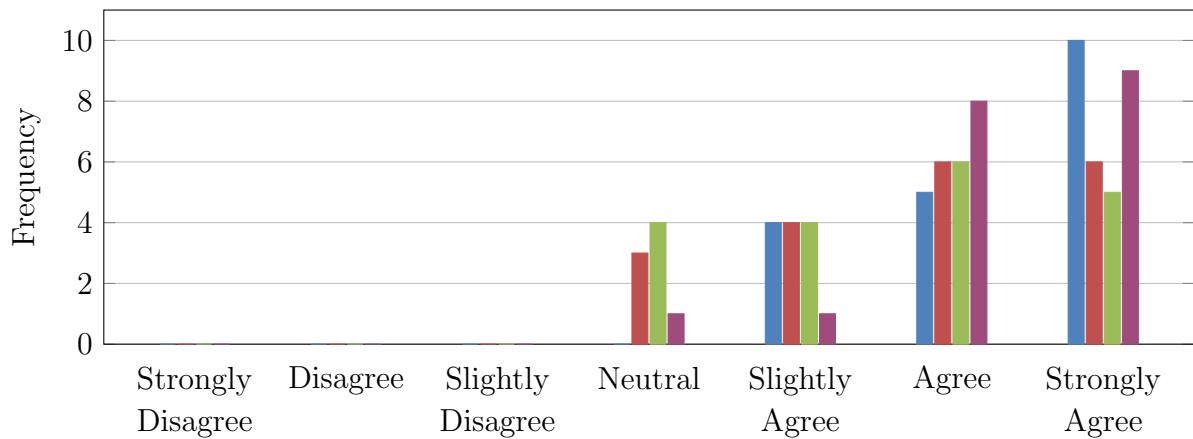
### 1.4.3 Nested Instructions

All participants found the addition of nested instructions with repeats preferable, with the majority strongly agreeing.



### 1.4.4 Active Turtles List

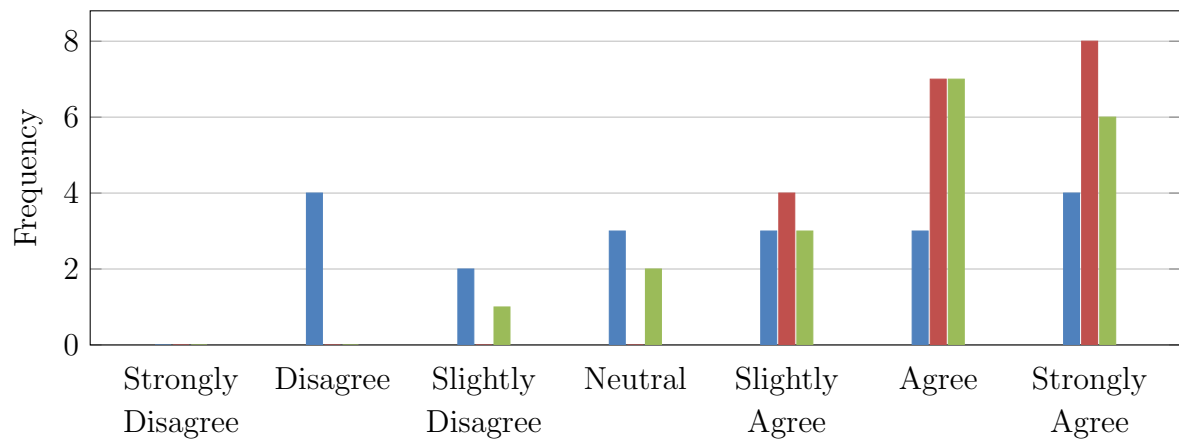
Statement	Mode	p-value
■ it is easier to tell if a certain turtle has been registered.	Strongly Agree	0.0000
■ it is easier to see where the active turtles are.	(Strongly) Agree	0.0011
■ it is easier to toggle the activation of turtles.in the correct place	Agree	0.0038
■ Overall, system 2 is preferable.	Strongly Agree	0.0000



The addition of a list of active turtles was found by all users to have a neutral or possitive effect on Excello.

### 1.4.5 Continuous Volume

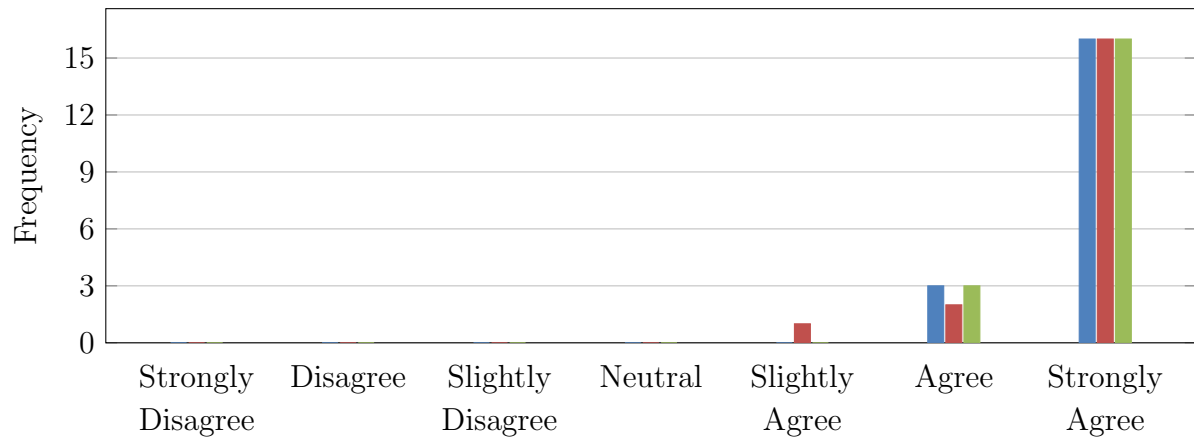
Statement	Mode	p-value
■ it is more intuitive how loud a note will be played.	Disagree Strongly Agree	0.6592
■ the volumes available are less limited.	Strongly Agree	0.0000
■ Overall, system 2 is preferable..	Agree	0.0003



There is no significant result for whether the ability to define volume in the range  $[0,1]$  is more intuitive. All users agreed that the volumes were less confined. However, only one user did not find this change preferable. Given that, it can be ommitted and the previous conventional dynamic markings used, this further supports the evidence that this addition was still successful.

### 1.4.6 Automatic Stepping

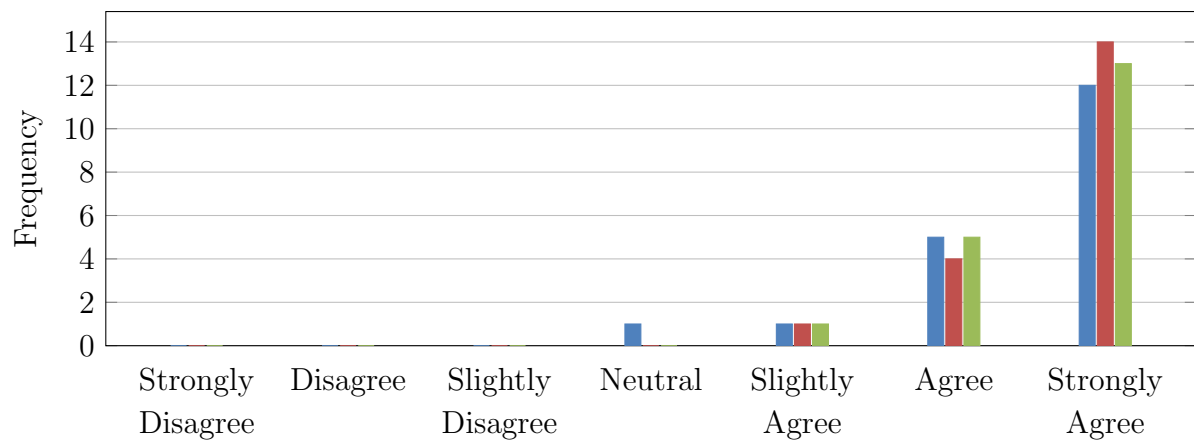
Statement	Mode	p-value
■ less mental work is required to write the turtle instructions.	Strongly Agree	0.0000
■ less work is required when more notes wish to be added.	Strongly Agree	0.0000
■ Overall, system 2 is preferable.	Strongly Agree	0.0000



The addition of this feature was particularly successful with 16 of the 19 users strongly agreeing the system was more preferable with automatic stepping available in the turtle instructions.

#### 1.4.7 Absolute Tempo

Statement	Mode	p-value
it is easier to tell what the speed instruction corresponds to	Strongly Agree	0.0000
giving an exact tempo (e.g. when transcribing sheet music) is easier	Strongly Agree	0.0000
Overall, system 2 is preferable.	Strongly Agree	0.0000



The initial design was changed so that turtle speed was defined in absolute cells per minute. All users found this to be an improvement.

Overall, the participatory design process was very successful. Multiple features were added to Excello to solve problems identified through formative evaluation sessions and longer-term feedback from users. There is significant evidence to suggest all added features were found to improve Excello.



## 1.5 Cognitive Dimensions of Notation

Figure 1.3 shows the percentage of time users identified carrying out the different cognitive activities [4] in Excello and in Sibelius. The Excello results are for 19 users with 12 users for Sibelius. Translation is a very important activity for both interfaces. Comparing the Excello results to Sibelius, there is more exploratory design but as users become more familiar with the system and the amount of existing Excello notation increases, modification and incrementation may become more important. Little time is spent searching for information in the notation in either.

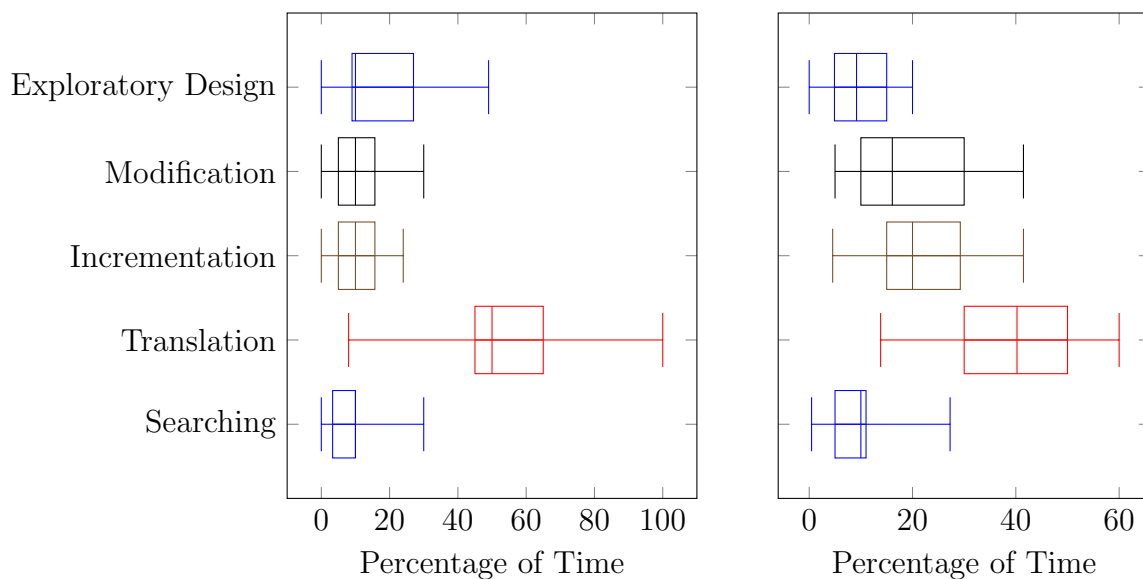
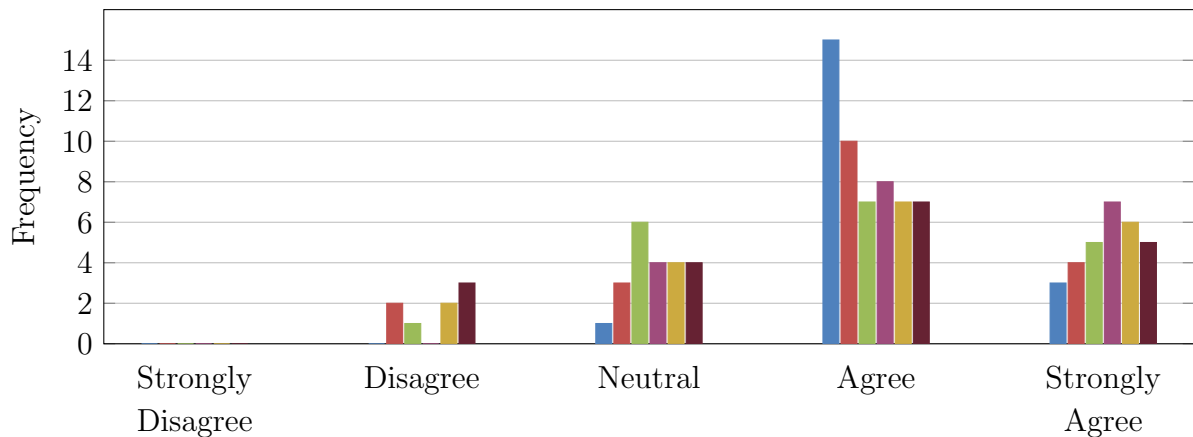


Figure 1.3: The percentage of time spent performing the different cognitive activities in Excello (left) and Sibelius (right).

A series of statements assessing the CDN of the interface were selected [2], these were assessed using a five-point (Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree) Likert scale. To verify the significance of the results, the data was combined into negative and non-negative categories and a chi-squared test performed. The questions asked, dimension being assessed, p-value from the chi-squared test and modal response are shown in table 1.1. The distribution of responses is shown in figure 1.5.

Statement	CDN	Mode	p-value
■ (a) The notation used (In Excello: notes /dynamics in cells and the definition of turtles) is related to the result you are describing (In Excello: Musical output)	Closeness of Mapping	Agree	0.0004
■ (b) Where there are different parts of the notation that mean similar things, the similarity is clear from the way they appear.	Consistency	Agree	0.0087
■ (c) You can add extra marks (or colours or format choices) to clarify, emphasise or repeat what is there already.	Secondary Notation	Agree	0.0020
■ (d) When you need to make changes to previous, work it is easy to make the change.	Viscosity	Agree	0.0004
■ (e) It is easy to see or find the various parts of the notation while it is being created or changed.	Visibility/Juxtaposition	Agree	0.0087
■ (f) If you need to compare or combine different parts, you can see them at the same time.	Visibility/Juxtaposition	Agree	0.0312

Table 1.1: Questions and results for testing the CDN of Excello



As these questions were also answered for the user's interface of choice, a comparison to Sibelius can be made. As the data does not meet the assumptions of the t-test [1], I performed a Wilcoxon matched pairs signed-ranked test on the 12 pairs by encoding the five possible responses as -2,-1,0,1,2. For all six questions, there is no indication that the answers for the two interfaces come from populations with different means.

### 1.5.1 Closeness of Mapping

Sibelius uses an established music notation as part of large professional software. As there is no significant evidence that the population means for Excello and Sibelius were different, this suggests Excello notation in spreadsheets has not compromised that close-

ness of mapping. This is helped by using an existing notation for defining the individual notes, the turtle instructions mapping to a movement through the grid, and by adjusting the speed argument to be an absolute, not relative, parameter. Being less familiar with staff notation, user 4 found the Sibelius's unintuitive.

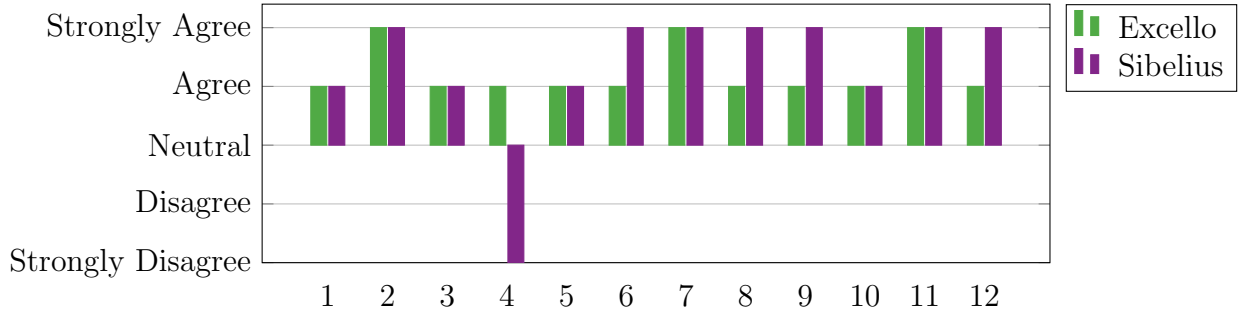


Figure 1.4: User responses for closeness of mapping for Excello and Sibelius from (a)

### 1.5.2 Consistency

As each cell and turtle can only cause one note to play at a time, consistency is maintained by building up from these building blocks. Excello keeps consistency with the existing Excel interface by sharing notations (e.g. A1:A5 for ranges) and using the existing formula editor. Within the turtle instructions the ability to use a number after instructions for repeats holds for both individual instructions and sequences. This all helps contribute to there being no significant result from the Wilcoxon test.

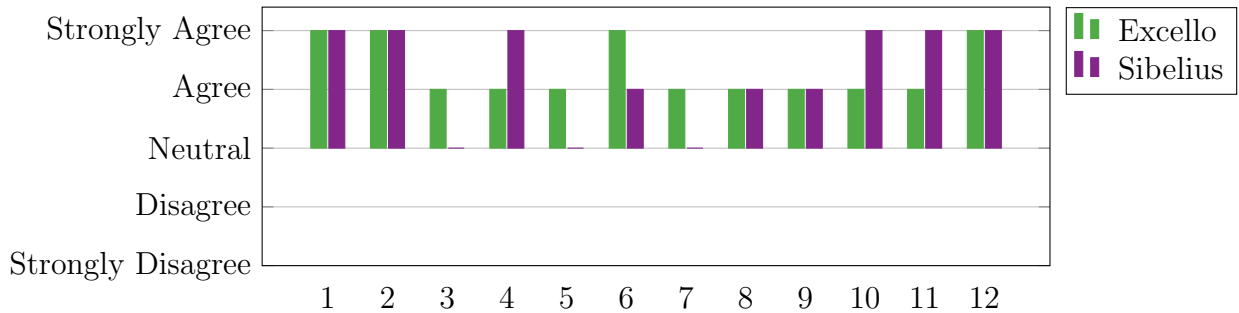


Figure 1.5: User responses for consistency for Excello and Sibelius from (b)

### 1.5.3 Secondary Notation

Given the time spent performing translation, secondary notation is particularly important [3]. Given that Excello abstracts time from the axes of the grid the distribution of parts is up to the user and cells can be used for arbitrary marks. As a result the existing features of Excel to format and group cells is already available. This is already utilised by the highlighting of notes and turtles. That there is no significant difference in population

means, this suggests that the spreadsheet paradigm can provide an equal secondary notation abilities to Sibelius, software already equipped with numerous ways to customise a score.

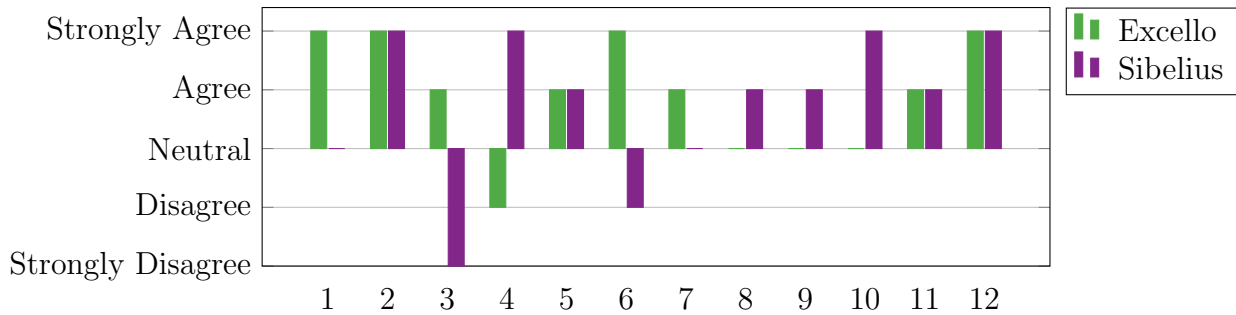


Figure 1.6: User responses for secondary notation for Excello and Sibelius from (c)

### 1.5.4 Viscosity

By allowing dynamics and octave marking to be ommitted and the ability for turtles to step forward automatically, there is low resistance to making additions and changes to the music. The button to toggle turtle activations dramatically reduces the actions required to turn turtles on and off. Furthermore Excel allows for the easy editing and movement of cells. No significant result in the Wilcoxon test suggests the interfaces have comparable viscosity.

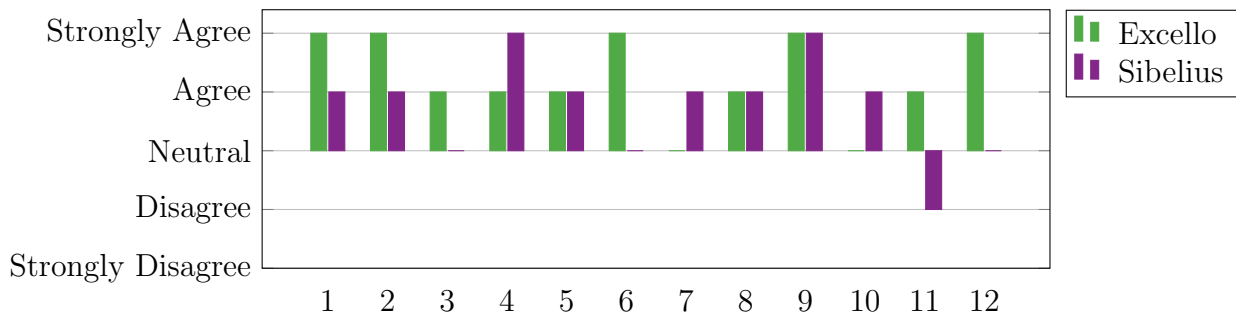


Figure 1.7: User responses for viscosity for Excello and Sibelius from (d)

### 1.5.5 Visability / Juxtaposition

For both questions there was no significant difference in population mean. This suggests that the spreadsheet interface can provide a similar ability to view components than Sibelius.

Whilst there was no significant evidence to suggest Excello outperformed Sibelius across these CDN, that it achieved similar results to Sibelius, an established music notation editing package, suggests that Excello is successful interface for writing music.

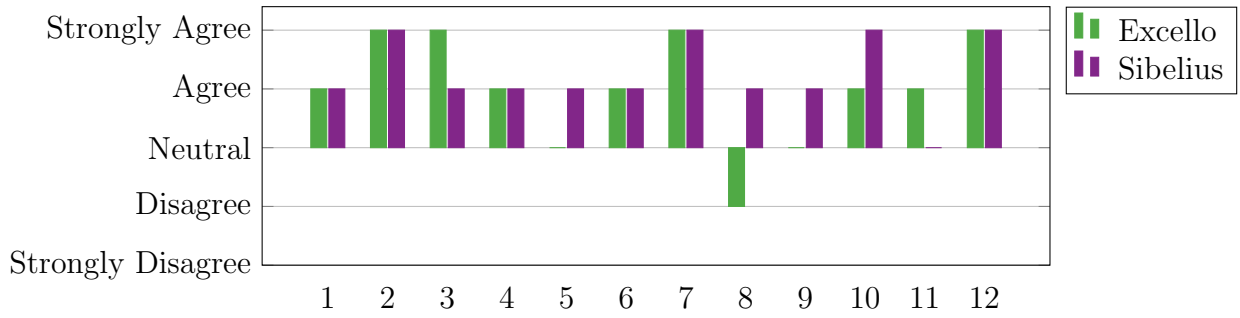


Figure 1.8: User responses for visibility/juxtaposition for Excello and Sibelius from (e)

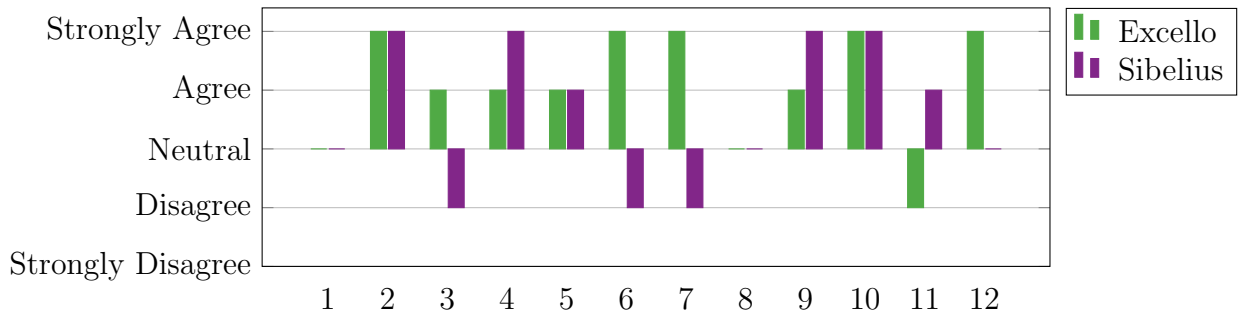


Figure 1.9: User responses for visibility/juxtaposition for Excello and Sibelius from (f)

### 1.5.6 Other Dimensions

If users are unfamiliar with the turtle paradigm, this may reduce the *role-expressiveness*. Turtles and notes are the only components of the spreadsheet and these are identified by highlighting. Whilst the Excel is very flexible and notes and turtles can be added in any order, adding an additional part may required many line insertions which increases the *premature committment*. The dual-formalism of the turtles and notes could create high *diffuseness* but the freedom of the user to lay these out allows this to be minimized as in figure 1.1. This also shows how representations can have goot *synopsie*, as the notes or turtles don't need to be examined to understand what is happening. However, this may come at the expense of *hidden dependencies* where it is not immediately clear which notes are triggered by which turtles. Volume is also dependent on the notes turtles play before it. But as a single cell could be played at multiple volumes, this is a tradeoff of this design decission.

As well as the  $\mathbf{m}^*$  notation decreasing *viscosity*, it also improves the *progressive evaluation* as turtles can be played before a whole part has been transcribed. The highlighting of cells also helps users receive more feedback. The ability to define a turtle and fill in the notes later also improves the *provisionality*. The use of  $\mathbf{m}^*$  also reduce the *hard mental operations* required. The chord input tool also removes the need to calculate the notes of a chord.

Spreadsheets are "an abstraction-hating system" [4], therefore little *abstraction* is provided by Excel, but the grouping of turtles in one definition and nested bracketing in

turtle movement instructions improve this. These features also provide good *legibility*.

## 1.6 Ethics and Data Handling

After ethics approval, the pilot session for the formative evaluation session was designed. Participants were provided with a consent form explaining the project and the format of the session. Participants had the choice to remain anonymous so they would not appear in acknowledgements, if so, all their data was labelled with a unique ID for participants to be able to use to request removal or anonymising of their data. All participant data was only seen by myself. Sessions were audio recorded with recordings types up after the session and then deleted. All participant data was also backed up on GitHub with the rest of the project but in an encrypted folder. After the pilot session (also performed for summative evaluation), the session was revised before continuing with the remaining sessions.

# Bibliography

- [1] Dwight Barry. Do not use averages with likert scale data, 01 2017.
- [2] Alan F. Blackwell and Thomas R. G. Green. A cognitive dimensions questionnaire optimised for users. In *PPIG*, 2000.
- [3] Alan F. Blackwell, Thomas Green, and Dje Nunn. Cognitive dimensions and musical notation systems. *Workshop on Notation and Music Information Retrieval*, 11 2000.
- [4] Thomas Green and Alan Blackwell. Cognitive dimensions of information artefacts: a tutorial. Technical Report Version 1.2, BCS HCI Conference, 1998.
- [5] Thomas Green and Marian Petre. Usability analysis of visual programming environments: A 'cognitive dimensions' framework. *Journal of Visual Languages*, 7:131–, 06 1996.
- [6] Chris Nash. Manhattan: End-user programming for music. In *NIME*, 2014.
- [7] S.M. Ross. *Introductory Statistics*. Elsevier Science, 2010.