Spem in Alium:
The Virtual Choir Project

James Haupt
April 28, 2003

- Table of Contents -

1.1 A Brief History and Introduction

Thomas Tallis was born c.1505 and died in 1585. He led a rather undocumented life in what more today is more commonly referred to as the Medieval Ages in music. Although much of Tallis' history is early unknown, we are able to gather that he lived in England, saw 4 different monarchs, and managed to survive by being flexible in both religion and with his music. ("Thomas Tallis – A short biography…")

One of Tallis' most famous compositions, the 40-voice *Spem in Alium* was created out of ingenuity, determination, and a challenge by the Catholic Thomas Howard, fourth Duke of Norfolk to produce a piece that rivaled Allessandro Striggio's (Italy) 40-part *Ecce beatum lautam*. However, that is only one version of the story. Another "says that Tallis wrote the 40 part motet for the 40th birthday of Queen Elizabeth I in 1573. The choice of text, from the Book of Judith, may have been to flatter the Queen, comparing her to the heroic Judith." ("Thomas Tallis – Spem in Alium…")

In either event, Tallis left a lasting impression on music for years, until approximately 400 years later.

1.2 Computers and Digital Music

With the advent of computers in the 1960s, and their eventual commercialization in the late 1980s, music has grown to b e able to accommodate beginners to even the most advanced musicians. Commercial products such as

Finale and *Digital Performer* allow anyone to compose their own pieces of music. Using this technology we're able to take something that was created nearly 500 years ago, and transform it to have an entirely different meaning.

Perhaps one of the greatest similarities between music of today and music of the middle ages is that is very difficult to get 40 people together to perform a piece, let alone each of them singing a separate part. Learning to sing a piece of that magnitude alone can be a chore, not to mention the fact that you're singing against 39 other voices, none of which are singing the same part as you. Using sampling technology readily available today, it is possible to remove the problem of teaching 40 people to sing different parts, as well as to get them all together at once.

1.3 Revival of the Part Book

A part book is a large book designed such that many people can read and sing from it at once. More commonly used in early music periods, this idea stemmed from the need to have many people reading the same music, but without the cost of giving each person their own book. This idea is carried over to the modern orchestra, where music is placed on a stand and one to two people read from the music at the same time.

While one could almost think that the part book has never really gone away, however, in the choral setting it has long since been forgotten. As paper/parchment grew cheaper during the baroque period, it was no longer

necessary to produce one large book for an entire section of the choir to read, and instead the large book evolved into smaller books for each singer to have. These books would however contain the music for the entire choir (as opposed to the part book which only contained their part), so that a person singing one part could observe what the rest of the choir is singing, and line up their own part to match the others. This has served the choir well for many centuries as popular music typically has only four parts: Soprano, Alto, Tenor, and Bass.

When we stop to consider what a modern book would look like with 40 parts however, it easily can be compared to a full orchestra conductor's score. Now imagine 40 people each attempting to sing from a conductor's score, everyone singing a different part. Let alone the ease of losing their places in the music, the possibility of changing choirs in the middle of singing (or resting for that matter) become very great, the same reason each person in an orchestra doesn't get a full version of the conductor's score. For this project I attempted to produce a similar situation.

While the original goal of this project was to digitize *Spem in Alium* (see Section 2.1), it was a small step away to produce part books. With the help of Prof. John Delorey's Finale class, we were able to create 8 individual scores, one for each choir.

2.1 Digitizing *Spem in Alium*

For this portion of the project I relied heavily on Coda Music Technology's *Finale 2002*. Actually taking *Spem in Alium* and putting it into the computer can be a bit of a daunting task, however, with the help of a MIDI compatible keyboard and using the HyperScribe tool makes the job a lot easier.

The next step after getting the entire piece into the .mus format was to have Prof. Delorey's Finale class go over it with a fine toothed comb to check for any mistakes. Once that was completed, they were able to break up the piece into its respective eight choirs and add words to create part books.

While these part books will most likely not be used for the actual performance, they will prove to be very helpful during the recording phase of the project.

3.1 Digitally creating the Choir

This is undoubtedly the longest phase in the entire project. From beginning to end this involves the recording of people singing in order to split up the individual notes to downloading each note onto the samplers so they can be played back in order. For ease of reference this large phase is split up into two smaller ones: the recording phase and the splicing phase.

3.2 Recording the Choir

The goal of this portion of the project was to get all 40 voices recorded onto the Power Macintosh using *Digital Performer*. While it may be easy to think of in a recording sense, there however are quite a few possibilities as to how to record. As this really hasn't been done before, there's no frame of reference and I'm left only to rely on Prof. John Delorey's helpful suggestions as to how things should be done.

One of the most prominent problems that presents itself when one goes to record a selection of music is how many people should be singing the part? There are really two possibilities here, either have one person per part, or having two people on a part. (Any more than two creates very noticeable tonal clusters, as the more people that sing together; the less likely they are to sound together).

Seemingly by chance (or perhaps some odd planning on Delorey's part), we wound up having 1 Soprano, 1 to 2 Counter-Tenors, 2 Tenors, 2 Baritones, and 2 Basses for part divisions. Whereas this could have changed at a moment's notice, I decided to keep this balance after recording the first choir.

Each choir received its own *Digital Performer* file which was then split up into the respective parts, and then further split up into the respective phrases for ease of splicing and identifying individual notes.

3.3 Splicing the Notes

Out of everything this project entails, this truly is where the most work is put in. After recording all 9726 notes in 139 measures, the task then becomes to splice those notes into 9726 individual files. When taking into consideration the fact that we're dealing with human voices, it becomes even more difficult.

Whereas for each note on an instrument there's a definitive beginning, middle, and end, it is much different with a human voice. Also unlike an instrument, it's impossible to use a prerecorded set of notes for multiple applications since words need to be taken into consideration. Hence the 9726 individual files. To begin we need to analyze what happens when someone goes to sing a tune. For example's sake, I will use the tune *Twinkle, Twinkle, Little Star*.

So let's start with the first word: Twinkle. Obviously we can break this down into any numerous different ways from Twin-kle, to Twi-nkle, however depending on how the person sings the word will determine how it should broken up. This example presents two possibilities: either the person singing sings a long 'n' so the word becomes Twinnnnkle, or sings a short n in which case the word becomes Twi-ngkle.

Now imagine that we want to be able to preserve the way the word sounds, regardless of how fast or slowly the tempo we take the piece at is. It becomes pretty apparent where the split should be made between the first and the second notes. In the first example the first note should end right before the k, and the second note should include the kle. In the second example however, we

want to end the first note before the ng, and then include the n sound with the

kle so the second note becomes nkle. If we look further into why it should be this

way, all we need to consider is if the first note is cut short what is the best way to

be able to preserve all of the consonants. This is my process of thinking for

splicing *Spem in Alium.*

A much broader view is evident in the e-mail I sent to the two sufficiency

students (Jaron Kuppers and Ben Apollonio) working on this portion of the

project:

```
I'll begin by starting at the beginning of a phrase.
This should be fairly obvious, as the first note
(whether it has a consonant at the beginning or not)
needs to be one soundbyte. The end of the note is where
the discrepancy happens as there are three
possibilities to what could occur: it ends on a vowel,
it ends on a hard consonant, or it ends on a soft
consonant.

If it ends on a vowel, you need to include that vowel
to its very end (right up to the next note).

If it ends on either type of consonant, you need to end
the soundbyte right before the beginning of that
consonant.

Granted this can be difficult as there are two singers
and they may not both say the consonant at the same
time, however, let's just make it a general rule that
the first person to say the consonant goes (so start
from there for the next note).

You can pretty much follow that for the rest of the
phrase until you get to the end, and whether it ends on
a consonant or a vowel. Again, if it's a vowel, leave
it as part of the last soundbyte. If it's a consonant,
make a soundbyte just for that consonant, so we can
play it on at the next beat (like a singer would hold a
note out until the next beat, and then put the
```

```
consonant on that beat as opposed to ending it before
then).

For the long phrases where just one vowel is held over
multiple notes, just start and end each soundbyte where
the change occurs (like singing one vowel after
another).
```

4.1 Future Logistics

At our current stage both Kuppers and Apollonio are still working on

splicing the notes. Once they have finished we will be able to download all of the

soundbytes onto samplers and begin to shape the phrases. A lot of the work will

now be involved in using *Digital Performer* to constantly output patch changes to

each sampler as each note changes, since each note needs its own soundbyte.

Once this is completed we can accomplish an actual performance.

4.2 The Performance

There is still a lot of decision making left to be done for this phase. Tallis

himself provided multiple different ways the choirs can be situated in a room to

produce different effects; everything from having all 8 choirs in a straight line to

having them around the entire room in a circle are presented. However,

something such as a circle becomes difficult if a director is needed to keep all of

the choirs together since it may be difficult to see the director at times. This is not

a problem since all choirs will be virtual and all that is needed is to position the

speakers and run speaker cable.

However, as an addition, Delorey would like to perform this with one live choir, adding an entirely new sound to the mix. Problems should arise (and easily corrected) with balancing the one live choir with the seven virtual ones.

4.3 Future Applications

While it is a large task to take one piece and record note by note into synthesizers for each parting order to perform only that one piece without the ability to change to another piece easily, perhaps in the future we could build a database of multiple syllables sung by different voices on different pitches varied enough to create a more versatile virtual choir. In its current state, this project only allows the piece *Spem in Alium* to be sung, however, it may become possible as technology advances to change pitches more on the fly and be able to have the choir sing more than one song.

5.1 Credits

This project is a spin-off of Prof. Fred Bianchi's Virtual Orchestra technology. Incorporating the same ideas to vocal synthesis and performance allow even the smallest choir of 5 to become a choir of 80.

Prof. John Delorey has been an immense help to me with suggestions and ideas as well as singing the counter-tenor part for the majority of the piece.

Jaron Kuppers and Ben Apollonio for their work during the splicing portion of the project.

Special thanks to Diane Jordan, Evan MacCarthy, Allan Sword, Tyler Ames, and Rick Richter for donating their time as well as letting me record and butcher their voices.

Also thanks to Jason O'Leary for his paper on recording and using the E-MU samplers that were made available to us through Fred Bianchi.

Works Cited

"Thomas Tallis (c.1505-1585) – A short biography & discography" – Sue Hickey

www.medieval.org/emfaq/composers/tallis.html 4/27/2003

"Grove Music" – Laura Macy www.grovemusic.com 4/27/2003

"Thomas Tallis – Spem in Alium – A Good-Music-Guilder Review"

www.good-music-guide.com/reviews/027_tallis.htm 4/27/2003