International Symposium on Performance Science
ISBN 978-94-90306-01-4

Computer assisted analysis and display of musical and performance data.

Stuart Pullinger[1], Nicholas Bailey[1], Jennifer Macritchie[1] and Margaret McAllister[3]

[1] Centre for Music Technology, University of Glasgow, UK
[2] Department of Composition, Berklee College of Music, USA

The coordinated storage of performance data in such a way that it can be used across multiple projects is problematic: general purpose systems which can store gestural, score and other performance data are not generally available.

Using data from current projects, we aim to provide a unified database which can store and present musical score alongside associated performance data and musical analysis. Using a general purpose representation language, Performance Mark-up Language (PML), aspects of performance are recorded and analysed. Data thus acquired from one project is made available to others. Presentation involves high-quality scores suitably annotated with the requested information. Such output is easily and directly accessible to musicians, performance scientists and analysts.

We define a set of data structures and operators which can operate on musical pitch and musical time, and use them to form the basis of a query language for a musical database. The database can store musical information (score, gestural data, etc.) and audio/video artefacts. Querying the database results in annotations of the musical score, potentially augmented with audio/video selected from stored performances.

Two demonstrations are provided: an analytically-based query and a performance-gesture-based one. In the former, dissonant notes/intervals are identified in a performance of a Bach two-part invention. The score is then graphically annotated to indicate the performers' mean inter-onset intervals in the neighbourhood of these features. In the latter, a score of a 19-ET microtonal song is displayed annotated with the deviation in the soprano's pitch from that notated.

The database is capable of storing musical score information and multimedia recordings, and cross-referencing them. It is equipped with the necessary primitives to execute music-analytical queries, and highlight notes identified from the score.

Keywords: Music Information Retrieval; Visualization; Performance Analysis; Score Analysis; Database

INTRODUCTION

In presenting work in performance science currently under way at our research centre, it has become increasingly obvious that tools available for the presentation and storage of experimental data relating to musical performance, its delivery and reception, are virtually absent. Many such tools are *ad hoc* and where standards exist, they are rarely supported by a rich variety of applications which permit generic capture, processing, storage and presentation of the data they represent. The need for a storage and interchange mechanism was evident and increasingly urgent. To that end, the construction of a general infrastructure was undertaken. Currently, three parts of it are in use. Performance Mark-up Language (Pullinger *et al* 2008), a superset of Recordare's MusicXML (Good 2001), permits the representation in a single file of the musical score, empirically determined performance attributes, and (still underdevelopment) musical structure information. Presentation utilities permit results of empirical studies, based on audio or performance data, to be presented in an immediately accessible way along side the musical score. This is an essential facility if, as is now generally believed (Expert Panel Discussion, ICMC2008), the information of most interest to performance scientists resides in the difference between the score and performance rather than the similarity. Tying the two together is the database described in this paper which is currently capable of producing results from queries extending over a large corpus of stored work, unencumbered by the assumptions of tuning system, and capable of performing efficient queries of the kind relevant to music and performance analysts, particularly where the music uses a diatonic scale. It is potentially capable of encapsulating results of such queries in real-time AV transports, permitting extracts of video recordings of performances to be accessed in response to such queries.

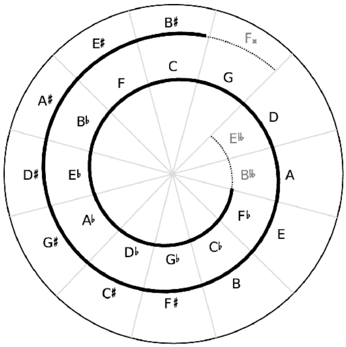
We begin by focusing on the data abstraction used to represent musical information, and the set of operations which are currently provided to implement searches over this data. Example queries are then presented, and the results are typeset in the context of the musical scores.

STORING MUSICAL INFORMATION

Music performance issues being addressed by the developers at the time the database's specification began to be formulated included the examination of pitch accuracy in the performance of microtonal music, specifically the Graham Hair's Microtonal Songs which use a 19-tone equal temperament. The notation remains unchanged, though enharmonic equivalences differ from those in 12-ET (Hair *et al* 2007). Previously existing methods of pitch representation vary from the frankly naïve, such as MIDI note number which is incapable of discriminating between an F-sharp and a G-flat, through to more sophisticated representations such as the Base-40 (Hewlett, 1992) and binomial (Brinkman, 1986) ones. It is indeed tempting to deploy such representation in a general musical database, but fixed-radix pitch class representations such as Base-40 do not generalise well when there are finer divisions of the scale, and although interval-invariant[1] exhibit values for which the pitch class is undefined. The binomial system is computationally challenging to use with scales which do not divide the octave into twelve because its separation of name class from pitch class means that pitch class representations mutate depending on the division of the scale. Regrettably, it seems advisable to define a new standard for our database's internal pitch representation, and the one that is chosen is based on the circle of fifths (Figure 1). This has advantages over other schemes in that:

- The representation can be extended to scales with quarter tones and more;
- Enharmonic-equivalent notes can be found in all diatonic scales;
- Most common scales (12ET, 19ET, 24ET and others) are simply represented;
- Simple for all scales constructed from a single size of fifth;
- Only slightly more complicated for other diatonic scales;

[1]The same interval is always represented as the same distance in pitch-class space



n	𝄫	♭	♮	♯	𝄪
F	-14	-7	0	7	14
C	-13	-6	1	8	15
G	-12	-5	2	9	16
D	-11	-4	3	10	17
A	-10	-3	4	11	18
E	-9	-2	5	12	19
B	-8	-1	6	13	20

Figure 1: The Spiral of Fifths and its Associated Number Line for Scales with One Division per Semi-tone.

- Distiguishes between enharmonic equivalent pitches – unlike MIDI &c;
- Interval invariant;
- Complete (no gaps/ambiguous pitches) – unlike Base-40 &c;
- One representation for all scales – unlike Binomial &c;
- Capable of representing diatonic microtonal scales.

A GENERAL CALCULUS FOR MUSICAL PITCH

Inspecting the table in Figure 1 and following the 12-tone chromatic scale we can see that enharmonically equivalent notes are 12 steps apart For example, C sharp $\leftarrow$ 8 and its enharmonic equivalent in the 12-tone scale D flat $\leftarrow 8-12 = -4$. A similar relation can be found for the 19-tone scale: A double-sharp $\leftarrow$ 18 and its enharmonic equivalent B flat $\leftarrow 18 - 19 = -1$, 19 steps apart. It is tempting to assume from this relation that in any scale of N divisions of the octave, the enharmonic equivalent of a note can be calculated as $k_{equivalent} \leftarrow k \pm N$. Unfortunately the generalisation is not quite so simple. To find the enharmonic equivalent of a note in terms of the next note name above it, it is raised by a major 2nd and then reduced by the number of chromatic scale steps in a major 2nd . For example to find the enharmonic equivalent of E sharp $\leftarrow$ 12 in the 12-tone scale, it is raised by a major 2nd ($12+2 = 14$) and reduced by the number of chromatic scale steps in a major 2nd ($14-(2\times7) = 14-14 = 0$) which results in F natural. This relation is expressed as follows:

$$k_{enharmonic} \leftarrow k \pm (i_{M2} - 7x)$$

where i_{M2} is the interval of a major 2nd and x is the number of chromatic scale steps in a major 2nd. Finding the enharmonic equivalent in terms of the name class above the current note equates to addition in the above equation and subtraction for the name class below. The relation expressed in the above equation ($k \pm i_{M2} - 7x$) resolves to $k \mp 12$ for the 12-tone scale and to $k \mp 19$ for the 19-tone scale. This allows the use of modular arithmetic to keep the result of any transformation within the gamut of the chromatic scale note names (i.e. excluding double-sharps and double-flats). Similar schemes can be used to represent other divisions of the scale, and, with extension, finer divisions than 19-ET. For example, we introduce a pitch denominator which represents the number of divisions per semitone. This allows us to represent scales requiring semi- and sesqui-sharps and flats.

The work we present here is based on the PostgreSQL open-source relational database, with extensions written in Python to provide the primitives necessary to perform musical queries.

The implementation stores pitch as a tuple $p := (k, z, o)$ where k is the spiral-of-fifths pitch class identifier, z the number of divisions per semitone, and o the octave (0 representing the octave containing middle C). Time is represented as a tuple $t := (n, d)$ where n, d are respectively the numerator and denominator of the duration in crotchets. A score note is represented as $s := (u, p, t, d)$ where u is a unique ID, p the pitch, and t, d are times representing onset and duration. A further data structure is provided to manipulate note groups of the form $g := (u, y, v, m)$ where u is a unique key, y is the group type, and v, m are ordered lists of, respectively, values and member IDs. The following pitch, time and (pitch-)interval functions are currently provided:

Ordinal Pitch Operators: lessThanPitch(p_1, p_2), greaterThanPitch(p_1, p_2), lessThanOrEqualPitch(p_1, p_2), greaterThanOrEqualPitch(p_1, p_2);

Pitch-Equality Operators: equatePitch(p_1, p_2), approxEquatePitch(p_1, p_2) (the latter ignores octaves in performing the comparison)

Intervallic Transposition Operators: addInterval(p, v), subtractInterval(p, v)

Interval Selection Operators: lowest(p_1, p_2), highest(p_1, p_2), getInterval(p_1, p_2)

Pitch Set Operators: scale(k, m) generates a scale in a given mode from starting at note k, elementOfPitchArray(k, K), approxElementOfPitchArray(k, K) tests for membership of note k in set of pitches K respectively honouring and ignoring the octave of k.

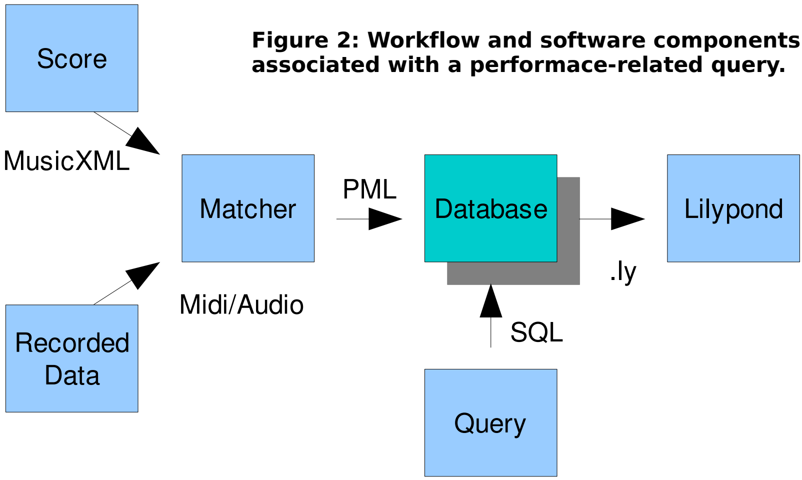
Time Operators: lessThanTime(t_1, t_2) etc., addTime(t_1, t_2)

Interval Functions: equateInterval(v_1, v_2) checks for intervals being exactly the same, approxEquateInterval(v_1, v_2) ignores octave offsets and equateIntervalClass(v_1, v_2) returns true if the intervals are of the same interval class (for example, major and minor 2nds).

CASE STUDIES

Methodology

Figure 2: Workflow and software components associated with a performace-related query.



The above diagram shows the work-flow associated with queries on the database which produce annotated output. The score, represented in MusicXML, is combined with recorded data (in both audio and MIDI format) and aligned using a score-performance matcher. This data is rendered in PML and used to populate the database. A query formulated in SQL extended as described previously is used to generate a table of results, and these in turn are converted automatically into a text file rendered in Lilypond, the musical score typesetter (Various, on-line resource).

1. Harmonic Intervals and Inter-onset Intervals in Bach's Invention no 1.

Performance information was captured by means of a MIDI Piano Bar (a device which rests on the end-cheeks of a grand piano and measures note onsets and releases optically), and synchronised automatically with the score of the piece represented in MusicXML to produce a PML representation of the performance. This information was used to populate the database. An SQL query was constructed to annotate a score with harmonic intervals, dissonant intervals being emphasised. The inter-onset intervals in seconds, presented graphically, are added to the annotation. Bar 4 is shown in Figure 3.

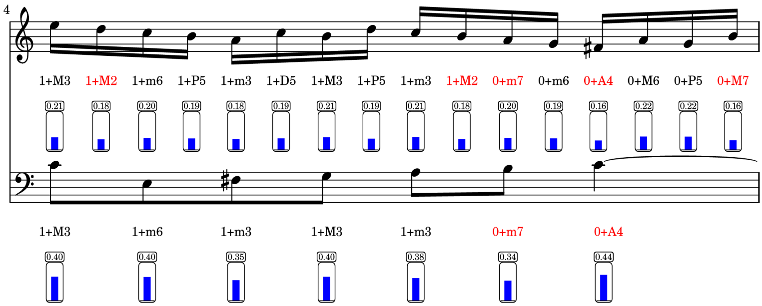
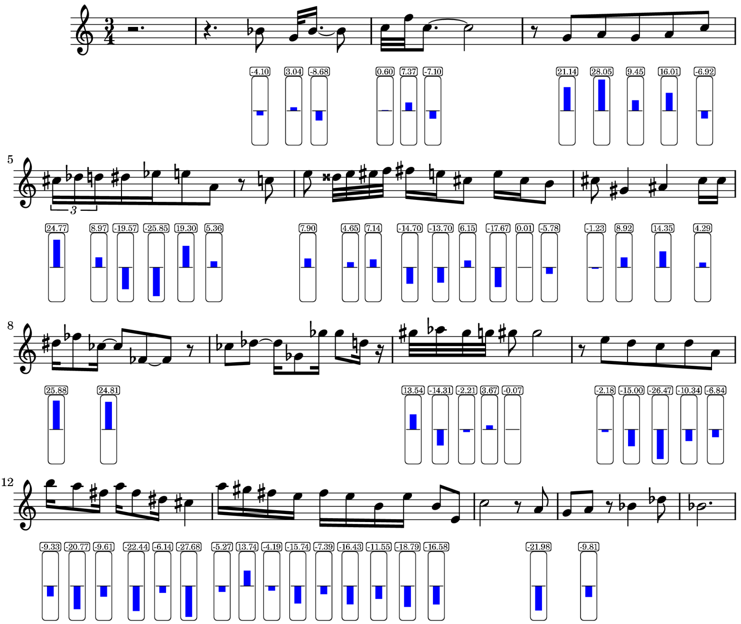


Figure 3: Bar 4 of J.S. Bach's Invention no. 1 BWV772, annotated with inter-onset intervals in seconds, and with harmonic intervals. Dissonant intervals are shown in red.

2. Pitch Deviation from the Notated Value in Hair's Microtonal Song, "Ash".

The pitch of the soprano part is acquired, segmented automatically using vector-quantisation techniques (Bailey *et al* 2008) and combined with the score to produce a PML representation of the performance. An SQL query was formulated graphically to annotate the score with the difference between the 19-ET pitch notated and the measured performance pitch in cents.

The performance shown is from an early practice session by an expert soprano before becoming accustomed to the 19-ET scale. Where gaps exist in the pitch data, the pitch tracking algorithm has produced unreliable results, or the notes have not reliably segmented and matched with the score.



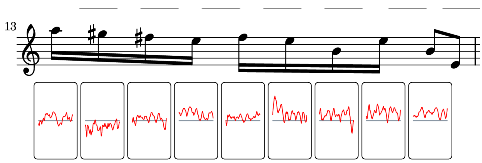


Figure 3:Microtonal Song, "Ash", by Graham Hair, with annotation showing difference between notated and performed pitch in cents (bars for sharp notes extend downwards). Detail (above right) shows pitch trajectory for each successfully matched note in Bar 13. Vertical scale is +/- 75 cents.

Bar 13 is further examined to establish the pitch trajectory for each note which was successfully matched. Note that where the performer inflected the note upwards, the notated-to-performed difference is negative, so a sharp note gives rise to a downward-extending graphic in the main part of the figure.

DISCUSSION, FUTURE WORK AND CONCLUSION

The pitch representation described here is computationally efficient, compact, and computationally extendible. Being based on a respected relational database, the system is robust enough to handle a large corpus of musical and performance data. The supplied primitives permit the flexible manipulation of performance data, and the production of results which, by close association of the score and performance data, is immediately accessible to professional music practitioners.

In future, it will be possible to wrap the results of queries resulting in the production of audio and video data in a transport such as the Xiph Corporation's ogg (Xiph Corporation). This format permits the transport of textual information, potentially allowing dynamic score updates alongside audio-visual playback when used with a suitable client.

The construction of the SQL queries alluded to in the previous section is currently an arduous task requiring considerable programming expertise. To produce the graphics shown here, the SQL code runs to approximately two pages of text causing space limitations to prevent its inclusion here. Full details of database extensions and the query language are to be found in (Pullinger 2009)

We are currently seeking performance collaborators wishing to use such a database facility in order to increase our experience of deployment and help with work towards a more elegant and user-friendly method of query design.

Acknowledgments

This research is partially supported by the EPSRC.

Address for correspondence

Nicholas Bailey, Department of Electronics and Electrical Engineering, The University of Glasgow, Rankine Building, Oakfield Avenue, Glasgow, Scotland G12 8LT, United Kingdom; Email: nick@n-ism.org

References

Nicholas Bailey, Douglas McGilvray, Graham Hair (2008) *Musically Significant, Automatic Localisation of Note Boundaries for the Performance Analysis of Vocal Music*. Proc Conf Interdisciplinary Musicology, Thessaloniki, 2008.

Alexander R. Brinkman (1986) *A binomial representation of pitch for computer processing of musical data*. Music Theory Spectrum, 8:44-57, 1986.

Expert Panel Discussion (2007) *Gesture Format Panel Discussion - ICMC 2007*

Graham Hair, Ingrid Pearson, Amanda Morrison, Nicholas Bailey, Douglas McGilvray, Richard Parncutt (2007) *The Rosegarden Codicil: Rehearsing Music in Nineteen-Tone Equal Temperament* Scottish Music Review, v1

Michael Good (2001) *Musicxml: An internet-friendly format for sheet music*, XML 2001 Conference Proceedings, Orlando, FL, December 2001.

Walter B. Hewlett (1992). *A base-40 number-line representation of musical pitch notation*. Musikometrika, 4:1--14, 1992.

Stuart Pullinger, Douglas McGilvray, Nicholas Bailey (2008) *Music and Gesture File: Performance Visualisation, Analysis, Storage and Exchange* Proc International Computer Music Conference 2008

Stuart Pullinger (2009) PhD Thesis, The University of Glasgow (in preparation)

Various Authors *Lilypond... Music Notation for Everyone* On-line resource. http://www.lilypond.org

Xiph Corporation *Ogg logical and physical bitstream overview*. On-line resource. http://www.xiph.org/ogg/doc/oggstream.html