# Mary had a little scoreTable* or the reacTable* goes melodic

Sergi Jordà
Music Technology Group
IUA, Universitat Pompeu Fabra
08003, Barcelona, Spain
sjorda@iua.upf.edu

Marcos Alonso
Music Technology Group
IUA, Universitat Pompeu Fabra
08003, Barcelona, Spain
malonso@iua.upf.edu

## ABSTRACT
This paper introduces the *scoreTable**, a tangible interactive music score editor which started as a simple application for demoing "traditional" approaches to music creation, using the *reacTable** technology, and which has evolved into an independent research project on its own. After a brief discussion on the role of pitch in music, we present a brief overview of related tangible music editors, and discuss several paradigms in computer music creation, contrasting synchronous with asynchronous approaches. The final part of the paper describes the current state of the *scoreTable** as well as its future lines of research.

## Keywords
Musical instrument, Collaborative Music, Computer Supported Collaborative Work, Tangible User Interface, Music Theory.

## 1. INTRODUCTION
### 1.1 To Pitch or Not To Pitch?
At the seminal NIME conference that took place in Seattle in 2001, Perry Cook stated that when building a new controller, one of the first things he would try to do is to play the simplest song such as "Mary had a little lamb" [5]. As a response to his statement, one of this paper's author defended a complementary approach [19], which was well applicable to FMOL, the instrument he was then presenting [9] and which is also clearly extensible to the one that came after, the *reacTable** [11]: instead of trying to replicate properties that can be very well handled by traditional instruments, we prefer to invent new instruments with the potential for creating music impossible to perform using traditional ones. As an example, our instruments allow to play simultaneously with timbre and form [10], controlling dozens of parameters of which pitch is not necessarily the favourite child.

Playing the "correct" notes is one of the least requirements for someone to be considered able to play a [pitched] instrument. When Max Mathews conceived the *Conductor*, the program that would be originally used in conjunction with the *Radio Baton* [14], believing that playing a different pitch from that written in a score would almost always be considered as a mistake, he chose to automatize this step using a predefined score. He therefore prohibited the performer any type of pitch selection. A decade later, Laurie Spiegel's interactive music software *Music Mouse* allowed the performer to indicate a tendency (e.g. higher or lower) leaving to the software the final selection of the correct notes [18]. When in 1997 one of the authors designed FMOL, an instrument that was conceived with the proselytist intention to introduce newcomers, possibly non-musicians, to more experimental and 'noisy' music, in order to avoid a "wrong note syndrome" that would inhibit the performer's experimentation and freedom of expression, he decided instead to minimize the importance of pitch. FMOL does neither fix nor correct pitch; it just dethrones it from its privileged position and turns it into merely another parameter, both at the control and at the perception level.

In any case, the prevalence of pitch as the primary variable in the music of many cultures [16], and the possibility of alternative but coherent and rich approaches, constitutes indeed a non trivial research topic that surpasses the purpose of this paper. This paper introduces the *scoreTable**, a new instrument – or perhaps a variation of an existing one – devoted to playing only pitch! We would like to stress though that this does not reflect a conservative turn into the authors' musical conception!



## 2. TOWARDS THE *SCORETABLE**
The *reacTable**, an instrument being developed by this team durimg the last three years, is build upon a tabletop tangible user interface [11, 12, 13]. Its performance paradigm (in which several simultaneous performers share the instrument's control by moving physical artefacts on the table surface and constructing different audio topologies) is inspired in the analog voltage controlled modular synthesizers [3]. Since each musical piece has to be constructed from scratch starting from an empty table, playing the *reacTable** is equivalent to building it. This establishes a continuum not only between composition and performance [4], but even between *lutherie*, composition and performance. Since this is combined with an extensive control on the lowest timbral level, the *reacTable** performs quite well at the poles of the musical spectrum: form and timbre. It does not excel however, at the intermediate level: it is hard to perform "Mary had a little lamb" on it. We consider this a feature more than a drawback. However, this topic is frequently raised when demonstrating the *reacTable** in general non-computer music contexts.

The idea of constructing a tangible music editor based on the *reacTable** know-how and technology (which includes the open-source computer vision engine reacTivision [1] and TUIO, a protocol for table-top tangible user interfaces [13]) was initially a humble one-week project aimed at showcasing different possibilities and applications of tabletop interfaces. The project was thus not meant to be very original neither too complicated. Several related instruments already exist and ours was not initially supposed to include outstanding innovation.

### 2.1 Tangible Sequencers
Enrico Constanza's *Augmented Musical Stave* [6] allows constructing simple melodies by manipulating rectangular blocks. The vertical position of the each object determines the

pitch whereas the duration of each note is predetermined by the objects themselves. The *Music Table* [2] enables the composition of musical patterns by arranging cards on a tabletop. A card's position on the axis running toward or away from the user determines the pitch of the note to be played, while its position from left to right determines its timing in a looping timeline. The *Music Table* allows also to change instruments by means of special instrument card, and to save patterns or phrases in special phrases cards by means of copy cards that behave as copy tools. The *Circular Optical Object Locator* (COOL) [8] is based on a hand-rotating platter on top of which opaque objects can be placed. Rather limited in is features, its radial configuration makes it more similar to the *scoreTable**.

All these implementations provide simple ways of "writing" music interactively by means of tangible user interfaces (TUI). None of them however, complements these physical artefacts with digital visual information, at least not directly; when they provide digital visual feedback they do it on a regular monitor separated from the table, loosing thus one of the key-features of TUI, which is the seamless integration of control and representation [20].

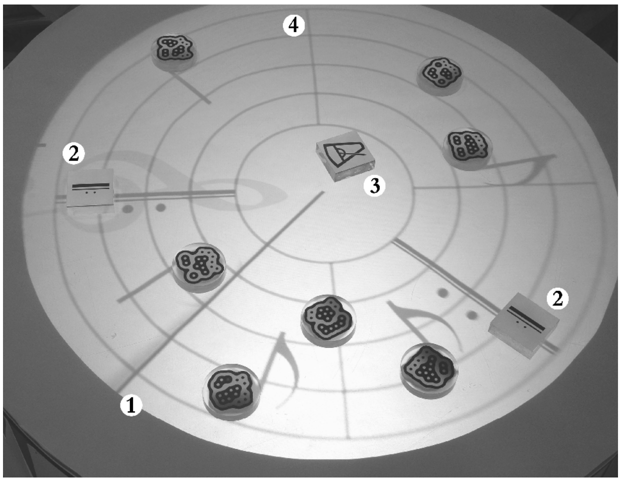


**Figure 1. The *scoreTable****

(1) radar sweep, (2) begin-repeat bar lines, (3) metronome, (4) bar separators

## 2.2 The basic *scoreTable**

The first quick and dirty *scoreTable** had basic and straightforward functionalities (all of which are still fundamental in the current version). It allows to position physical pucks (the same ones from the original *reacTable**) in a circular looping stave. A radar sweep rotates triggering the corresponding note (by means of the computer internal MIDI synthesizer) each time it passes a note puck. The pitch of each puck is controlled by its vertical position on the stave (its distance to the centre of the table) whereas its angular position, determines the onset time of the event. In this first version, by using a MIDI piano sound without sustain, we avoided note duration considerations and problems. Furthermore, rotating a special *Metronome* object changed the angular speed of the moving radius and thus the tempo. As shown in Fig. 1, visual and sonic feedbacks make this basic setup completely self-explainable. The perceived affordance of the first *scoreTable** (using Normans' approach to the term [15]), is clear for anyone who has ever seen a musical score. Playing with this simple model was, however, more exciting than expected; perhaps because writing music is normally a non-real-time activity! These and related considerations persuaded us to further explore the *scoreTable** potential. Its current state is described next.

## 2.3 Real-time Tangible Music Writing

The *scoreTable** uses the same physical pucks as the *reacTable**, which come in six different shapes. On the *scoreTable**, circular pucks are used for placing notes on the staff; squares are used for all types of non-note objects which have also a place on the staff (such as *begin-repeat* bar lines for the creation of loops) (cfg. 2 in Fig. 1).; cubes are used as 6-sides program changes, and the remaining ones (pentagons, rounded squares and domes) are used as different types of tools. Several pucks come in four different colours (RGBY) allowing four part writing. Colours are currently used in circular pucks (notes) and also in some "voice oriented" tools such as transposition.

Some tools are global, affecting the whole table independently of where they are positioned. They usually control only one parameter, which can be changed by rotating the corresponding puck. Some global tools are: *Metronome* (cfg. 3 in Fig. 1), *Number of Bars*, *Key Signature, Time Signature* or *Temporal quantization*. In this sense, because it was originally conceived as a tool for showing music notation to kids, musically speaking, the current *scoreTable** implementation is quite traditionalist and still intended for tonal music. *Key* and *Time* signatures changes are obtained by rotating their respective pucks and are instantaneously reflected on the table display. *Temporal quantization* quantizes notes onsets and durations according to a selected value. This is summarized in Table 1.

**Table 1. A summary of the *scoreTable** Objects and Actions**

| **Local controls** (score position dependent) | |
|---|---|
| **Notes** | Radial position → note pitch<br>Angular position → onset time<br>Color (RGBY) → instrument or track<br>Rotation → note duration |
| **Repeat bar lines** | Create loop regions on the fly<br>Angular position → Begin/end repeat time |
| **Global controls** (score position independent)<br>Usually control only one parameter that can be modified rotating the puck | |
| **Metronome** | Global table tempo |
| **Number of bars** | Determines the number of bars for one complete tour [1-8] |
| **Time Signature** | Changes the global time signature |
| **Temporal quantization** | Quantizes notes onsets and durations according to a selected value, which is a subdivision of a quarter note (or no quantization) |
| **Key signature** | Changes the global key signature (or no signature) |
| **Transposition** | Transposition (can be either *tonal* or *real*) |

## 2.4 Connecting to the *reacTable**

The *scoreTable** can operate by itself, sounding through an internal or external MIDI synthesiser (using the computer soundcard or sending it to the MIDI OUT port). It can also be connected to the *reacTable** using either MIDI directly, or sending OSC messages [21] through an Ethernet connection. In any case, four (RGBY) pucks on the *reacTable** receive pitch control information from the *scoreTable** and distribute it over the branches, according to the regular *reacTable** topological rules (see Fig. 4). In a complementary "Drum Machine Mode" each different note of the *scoreTable** can trigger an independent puck of the *reacTable**. In this case, the information is pitch-less and allows to control several independent *reacTable** branches directly from one *scoreTable** voice. This permits the control of complex and precise rhythmical structures difficult to obtain with the *reacTable**. This connection can also be used for sending other types of information, not necessarily pitched. Now that finger tracking is finally available on the *reacTable**, we plan to use this feature on the *scoreTable** for allowing users to draw envelopes of continuous time-based data. In this more flexible setup, the *scoreTable** fully becomes the time line of the *reacTable**.

# 3. FUTURE WORK AND CONCLUSIONS

## 3.1 Current Problems and limitations

We plan to overcome several "traditionalists" conditions, like the obligation of writing tonal music, but the *scoreTable** major limitations come from its hardware dimensions and technology. These will be harder to surmount. The size of the table (85 cm diameter) and the size of the pucks (5 cm diameter) are conditioned by the resolution of the camera and the computer vision engine we developed, *reacTivision* [1]. This conditions, on its turn, the size of the lines of the stave and leaves little room for additional control zones on the table surface.

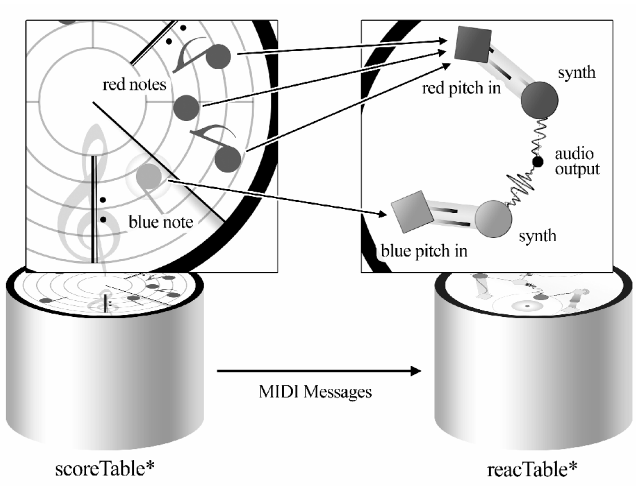


**Figure 4. Connecting the *scoreTable** to the *reacTable****

## 3.2 Virtual Operations

We are working on a new type of tools which will allow the generation of additional virtual material (i.e. notes that appear on the staff without associated pucks) as a result of the manipulation of physical notes. We are currently working on two operations, *Copy & Paste* and *Kaleidoscope*, but the list is not meant to be closed. These actions are more complex and require multidimensional control, possibly using more than one puck and sensing both the position and the orientation of each. They also constitute one of the more experimental and interesting interaction research topics of the project.

*Copy & Paste* tools are two-sided. One side allows to copy and store a sequence of physical notes, while the other allows pasting the contents (as is, transposed, augmented or diminished) in different parts of the score. *Kaleidoscope* operations on their turn, allow the creation of virtual notes on-the-fly. This is done by applying different types of symmetries to a slice of the score containing physical notes. *Kaleidoscope* will permit real-time control of musical set theory operations such as retrograde, rotation, inversion, transposition, multiplication, augmentation, diminution and combinations of them [7, 17]. *Copy & Paste* and *Kaleidoscope* will also be combined with paint pots, for cross-voice manipulations (e.g. copying the notes of a red voice fragment and pasting them to the blue voice).

## 3.3 The *scoreTable** and the *reacTable**: research on Tangible Musical Interfaces

"Traditional" instruments (acoustic, electric or electronic) as well as many non-traditional interfaces or controllers, force the performer to remain responsible, all the time, for all of the musical actions and nuances. This type of performance can be considered as the "synchronous musical activity" per excellence. On the other extreme, the *sequencer paradigm,* which still remains the most popular model of digital music creation, even if it typically incorporates some real-time actions, is mostly based on asynchronous interaction. A big mass of amateur musicians as well as professional composers and producers use a pool of standard sequencing software which try to melt, more or less seamlessly, the millennial model of the music writer with the ubiquitous and pervasive trends of the last twenty years of human computer interaction (WIMP, Drag & Drop, Copy & Paste…). The *scoreTable** is an odd hybrid that retains aspects of the "traditional" musical instrument (it is designed to be played in real-time, for "writing music performance"), while maintaining some typically asynchronous WIMP actions such as "cut & paste".

Using the TUI terminology introduced by Ulmer and Ishii [20], we can say that the *reacTable** is a relational system; it uses a homogeneous space and its topological properties are only defined by the relations between the objects on its surface, according to a building block strategy. The *scoreTable** follows a spatial approach instead; the positions where objects are placed determine their values and their functionalities. These objects can be on their turn, not mere tokens, but also *containers* (the bindings between the objects themselves and the digital information they convey becomes dynamic) or *tools*, which allow manipulating and changing the properties of other objects. In this sense, the *scoreTable** model is more "conventional" than the *reacTable**. It may also permit more "classic" research topics in Human Computer Interaction using TUIs, which could not be confronted in the *reacTable**. We thus believe that further research and brainstorming in "real-time tangible music writing" can thus bring some interesting results or ideas in music sequencing as well as in more generic and well-established human computer interaction areas.

From a musical point of view it is yet unclear if the *scoreTable** will be helpful for teaching musical notation to children, which was one of the first naïve assumptions, and something that has not been extensively tested yet. Still, it is already very fun to play and it promotes a very tight collaboration between the performers sharing the table surface. More toy-like, by itself, the *scoreTable** will probably not be either as compelling musical instrument as the *reacTable**, but it is our belief that a deeper understanding of all the concepts mentioned in this paper will lead to a positive cross-fertilization between both systems. By extension, the current parallel development of both projects, so related but so conceptually different, allows us to gain a deeper understanding of tangible user interfaces, their possibilities and their drawbacks.

# 4. ACKNOWLEDGMENTS

The *reacTable** team is currently constituted by Sergi Jordà, Martin Kaltenbrunner, Günter Geiger, and Marcos Alonso, who is also in charge of the *scoreTable**. Former members include Ross Bencina, who made several crucial contributions to the computer vision component and the OSC infrastructure and interns Ignasi Casasnovas and Gerda Strobl. This project is partially funded by the EU-FP6-IST-507913 project SemanticHIFI.

# 5. REFERENCES

[1] Bencina, R., Kaltenbrunner, M. & Jordà, S. (2005). Improved Topological Fiducial Tracking in the reacTIVision System. PROCAMS 2005, IEEE InternationalWorkshop on Projector-Camera Systems.

[2] Berry, R., Makino, M., Hikawa, N. & Suzuki, M. (2003). The Augmented Composer Project: The Music Table. Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03).

[3] Chadabe, J. (1975). The Voltage-controlled synthesizer. In John Appleton (ed.), *The development and practice of electronic music*. New Jersey: Prentice-Hall.

[4] Chadabe, J (1984). Interactive Composing: An Overview. *Computer Music Journal, 8(1)*, 22-28. Reprinted in Curtis Roads, ed., *The Music Machine*. Cambridge, MA: MIT Press, 1989.

[5] Cook, P. Principles for Designing Computer Music Controllers, NIME 2001, *ACM CHI 2001 Workshop on New Instruments for Musical Expression*, Seattle, April 2001.

[6] Costanza, E., Shelley, S. B. & Robinson, J. (2003). Introducing Audio D-Touch: A Tangible User Interface For Music Composition And Performance. *Proc. of the 6th Int. Conference on Digital Audio Effects* (DAFX-03), London, UK, September 8-11, 2003.

[7] Forte, A. (1973). Structure of Atonal Music. New Haven, Yale University Press.

[8] Hankins, T., Merrill, D. & Robert, J. (2002). Circular Optical Object Locator. Proceedings of the 2002 Conference on New Instruments for Musical Expression (NIME-02), Dublin, Ireland, May 24-26, 2002

[9] Jordà, S. (2002). *FMOL*: Toward User-Friendly, Sophisticated New Musical Instruments. *Computer Music Journal*, 26(3), 23-39.

[10] Jordà, S. (2005). *Digital Lutherie: Crafting musical computers for new musics performance and improvisation*. PhD. dissertation, Universitat Pompeu Fabra, Barcelona.

[11] Jordà, S. Kaltenbrunner, M. Geiger, G. & Bencina, R. (2005). The reacTable*. Proceedings of International Computer Music Conference 2005; Barcelona, 579-582.

[12] Kaltenbrunner, M. Geiger, G. & Jordà, S. (2004). Dynamic Patches for Live Musical Performance. In Proceedings of International Conference on New Interfaces for Musical Expression; Hamamatsu, Japan.

[13] Kaltenbrunner, M., Bovermann, T., Bencina, R. & Costanza, E. (2005). TUIO: A protocol for table-top tangible user interfaces. 6th International Gesture Workshop, Vannes 2005.

[14] Mathews, M. V. (1991). The Radio Baton and the Conductor Program, or: Pitch, the Most Important and Least Expressive Part of Music. *Computer Music Journal,* 15(4), 37-46.

[15] Norman, D. A. (1999). Affordances, Conventions and Design. *Interactions* 6(3), 38-43, May 1999, ACM Press.

[16] Pressing, J. (1990). Cybernetic Issues in Interactive Performance Systems. *Computer Music Journal, 14(1)*, 12-25.

[17] Rahn, J. (1987). *Basic Atonal Theory*. New York: London: Schirmer Books; Collier Macmillan.

[18] Spiegel, L. (1987a). *Operating Manual for Music Mouse: An Intelligent Instrument*. NY: Retiary.org.

[19] SIGCHI Bulletin, March/April 2002.

[20] Ullmer, B. & Ishii, H. (2001). Emerging Frameworks for Tangible User Interfaces. In *Human Computer Interaction in the New Millenium*, John M. Carnoll (Ed.), Reading, MA: Addison-Wesley, 579-601.

[21] Wright, M., Freed, A. & Momeini, A. (2003). OpenSound Control: State of the Art 2003. In *Proceedings of the 3rd Conference on New Interfaces for Musical Expression* (NIME03), Montreal, Canada, 2003.