Abstract

In this paper, we investigate the present state of Expressive Notation Package (ENP). ENP is a musical notation library realized inside PatchWork (PW). PW, in turn, is a general-purpose visual language with an emphasis on producing and analyzing musical material. The current features of ENP are discussed with the help of some musical examples.

1 Introduction

Along with commercial music publishing programs there have been several attempts to create systems capable of representing complex musical data - such as Common Music Notation (CMN, Schottstaedt) and PW Rhythm-Editor (RTM, Laurson 1996) - within LISP-based compositional environments. While being useful in some respects, they still lack many properties required by a professional notation package. When considering user-interface issues the commercial notation programs have a clear edge. For example CMN lacks a graphical interface and allows textual input only. The RTM editor is mainly meant to represent musical raw material and thus the editing capabilities are limited.

Nowadays several commercial notation programs allow the user to control the system via scripting languages. Often these scripting capabilities, however, are far too limited for compositional use. Another disadvantage is that the user must learn a new programming language (ManuScript™ in Sibelius7 or C++ in Finale) to be able to use this feature.

The purpose of ENP (Kuuskankare and Laurson 2000) is to provide professional notational capabilities, a graphical user interface and powerful object structures for compositional use. ENP is aimed at displaying scores using the traditional western notation. ENP is programmed with LISP and CLOS thus being extendible and open. It also provides full access to the musical structures behind the notation allowing ENP to be controlled algorithmically. Furthermore, ENP provides both standard and user definable expressions.

ENP has already been used to describe musical data for constraint-based applications (Laurson and Kuuskankare 2000) and to produce control information for several model-based instruments (Laurson et al. 1999 and 2001).

2 ENP

The graphical user interface allows musical objects to be edited with the mouse. The performed operation depends on the selected object. Notes and chords can be transposed and expressions can be repositioned or reshaped. Furthermore, all objects and their properties are editable through specialized editors. Expressions containing for example break-point functions can be edited with a break-point-function editor.

Musical data can be represented in different modes. Mensural-mode is used when working with traditionally notated metric music (figure 1). This example shows also some ENP specific expressions containing break-point-functions.

Figure 1. A mensural-mode example. J.S. Bach: Prelude.
Non-mensural-mode is used when writing music that is improvisatory and where the start times and durations of events are not exactly defined. A non-mensural notation example is given in figure 2 (above). This passage was entered as a sequence of chords with start times and durations given in milliseconds. The beaming information is not provided automatically by the system. The user can add the beams afterwards by hand. In this mode notes can be freely grouped and beamed to create the desired rhythmic structures. The user can drag the notes horizontally to edit the timing information (both start times and duration).

The third mode, aleatoric-mode, is suitable when working with scores that are written in the style of aleatoric counterpoint. In this mode, musical events are displayed as groups surrounded by boxes that, in turn, can be positioned relational to each other. These groups can internally be written in either mensural or non-mensural style (an example of this mode can be found in figure 3).

3 ENP Expressions

ENP provides a comprehensive collection of standard tempo-, dynamics- and articulation-symbols. Furthermore, it includes a set of non-standard notational attributes that can be used, for example, to describe control information for physical instrument models. Expressions can be applied either to a single note or to a group of notes.

Standard expressions include articulations, tempo indications, lyrics and so on. These have typically a more or less established graphical representation.

Non-standard expressions include special objects like groups. Groups are entities containing one or more musical objects. They can be used to give a set of objects a common identity. Groups can overlap freely. Thus, groups are useful in representing structures other than the ones provided by the beat structure.

Every expression is attached to some musical object or to a group of objects. This allows the expressions to move with the related objects. Once an expression is applied, ENP takes care of the correct placement of the musical symbols according to the notational conventions (Read 1982).

New expressions can be created through inheritance using a simple protocol. ENP builds automatically the needed user interface tools for any user created expressions.

4 ENP Examples

Figure 3 (above) gives a complex example that demonstrates some of the notational capabilities of ENP.

The first beat of the figure contains three expressions. First, there is a tempo expression. It can be freely positioned inside the measure by dragging the tempo-marking symbol. The second expression is a slur. The slur can be named in order to distinguish between slur objects or between different types of slurs (articulation or phrase). Third, we have a crescendo. It is a relatively complex object since it contains internally a break-point-function (figure 4). In addition, the initial and final dynamic levels can be defined symbolically (from \textit{mf} to \textit{ff} in this case).
The second beat of the example contains a chord with an accent and a 'let vibrate' expression. When applied to a chord the latter expression dynamically adjusts its graphical representation according to the size of the chord.

The next two beats contain some special note-heads (clusters).

In the beginning of the second line, we can see a special beaming ('feathered beam'). In ENP such ‘accelerando/ritardando’ rhythms are given as proportional durations according to which the beat is divided. ENP takes care of the graphical representation automatically.

The example ends with a gesture marked inside a repetition box.

Next, we give a transposed version of the previous example (figure 5). Here we can see how ENP handles automatically the correct placement of the musical symbols. For example, when a note is transposed all the expressions attached to it are automatically moved to their corresponding positions according to a predefined set of rules.

To conclude this paper we will present a part from a complete orchestral score created with ENP (see Appendix). This is a page from the piece called Arena by the Finnish composer Magnus Lindberg. The example given demonstrates some of the important concepts behind ENP. These include automatic spacing of the score, automatic positioning of the expressions and the possibility to export the score in PostScript.

5 Acknowledgments

This work has been supported by the Academy of Finland in project “Sounding Score - Modeling of Musical Instruments, Virtual Musical Instruments and their Control”.

References


