

Music Description and Processing: An Approach Based on Petri Nets and XML

Adriano Baratè
*Laboratorio di Informatica Musicale (LIM),
 Dipartimento di Informatica e Comunicazione (DiCO)
 Università degli Studi di Milano
 Italy*

1. Introduction

Music description and processing require formal tools which are suitable for the representation of iteration, concurrency, ordering, hierarchy, causality, timing, synchrony, non-determinism. Petri Nets are a tool which allows to describe and process musical objects within both analysis/composition and performing environments. To accomplish this objective, a specific extension known as Music Petri Nets was developed.

2. Music Petri nets

This Petri Net formalism can be applied to music field by associating music objects to places and music operators to transitions.

According to the definition in (Haus & Rodriguez, 1993), a *music object* may be anything that could have a music meaning and that could be thought as an entity, either simple or complex, either abstract or detailed. Such entity will present some relationship with other music objects. In a Music Petri Net, when a place containing an object receives a token, the music object is executed, i.e. played. To understand the following examples about transitions' behaviour, two simple music objects are shown in Figure 1 as notated fragments.

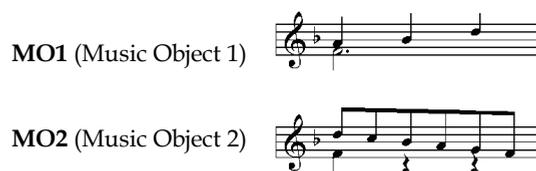


Fig. 1. Two simple music objects: MO1 and MO2

In Music Petri Nets the role played by transitions is very important: they determine – together with tokens – the evolution of the net. In our extension of Petri Nets transitions can have associated *music operators*. A transition without an algorithmic behaviour is considered having a null operator associated. When no music operators are associated, transitions are

Source: Petri Net, Theory and Applications, Book edited by: Vedran Kordic, ISBN 978-3-902613-12-7, pp. 534, February 2008, I-Tech Education and Publishing, Vienna, Austria

only devoted to net evolution. Their role is dropping tokens from input places and adding tokens to output places, such as in common Petri Nets.

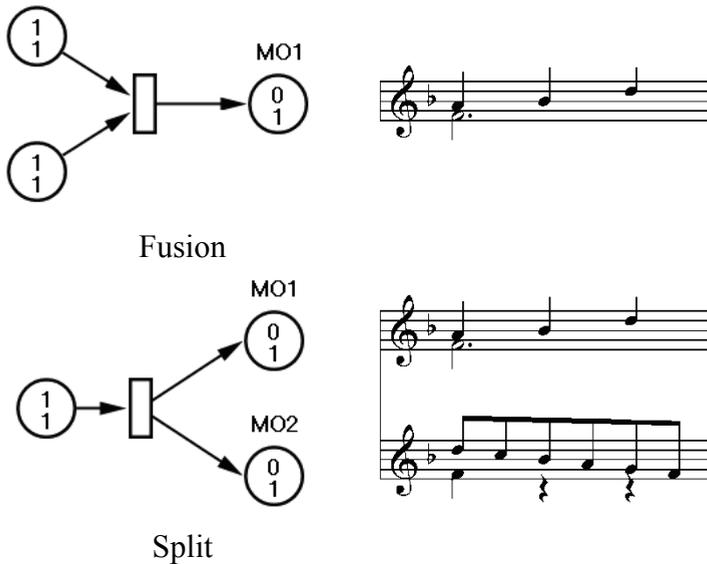
As stated before, when a token arrives at place with an associated music object, this object is played. In Music Petri Nets, the temporization of the execution is achieved considering the durations of the music objects (eventually) associated to the places. When a place receives one or more tokens from incoming transitions, the (eventually) associated music fragment is executed, and the tokens are blocked in the place (i.e. they cannot be transferred to outgoing transitions) until such execution is completed.

An example is provided in Figure 2, where the music object MO1 is associated to the left place with the same name, and MO2 is associated to the right one. The first measure is played when a token arrives in MO1, causing its execution. Then, only when the entire music object is played, the token is free to leave the place, and in fact it is moved to the subsequent one, originating the juxtaposed execution of the second measure. The overall result is noted in the score.



Fig. 2. The *sequence* structure

Various music structures can be created even by using transitions *without* music operators. In Figure 3 five simple nets illustrate respectively a fusion (from two objects to one object), a split (from an object to two objects), an alternative (a non-deterministic choice between two objects), and a joint structure (a logical connection between two objects).



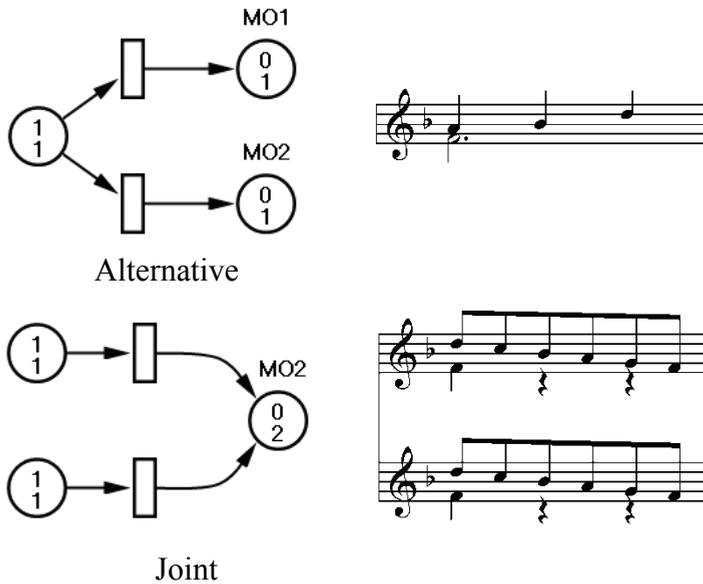


Fig. 3. Some PN structures and the corresponding executions

We have said that transitions might have associated music operators. When a music operator is specified, its purpose is applying an algorithm to change input objects (i.e. objects coming from input places), and then passing the transformed objects to output places. Typical operators associated to transitions reflect common music operators, such as inversion, retrogradation, and transposition. For example, Figure 4 shows the application of the last mentioned operator. In this net a music object is associated to the place MO3, while the right place has no associated objects. When the transition fires, it receives MO3 in input, a transposition is performed and the modified music fragment is passed to the outgoing place, that executes the new object.

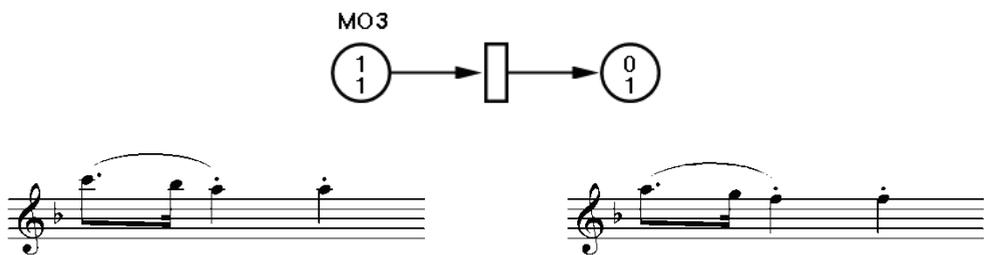


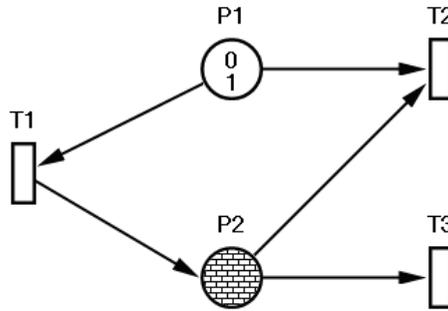
Fig. 4. Music Object 3 before (left) and after (right) the transposition

2.1 Extensions

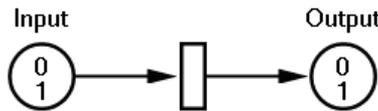
In Music Petri Nets some basic extensions are considered. Since this formalism is used to represent the structure of music pieces, together with its hierarchies, the natural choice is to include the *refinement* as a simple morphism mechanism. With this extension, deeper music

analyses can be integrated in subnets, permitting a better comprehension and decreasing the net complexity.

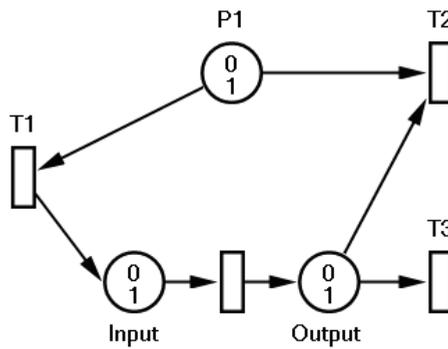
An example of refinement is presented in Figure 5. It must be noted that a node with a subnet must be of the same type of the subnet's input and output nodes, to achieve the expansion of the entire net.



a) The main net



b) The subnet P2



c) The expanded net

Fig. 5. An example of refinement

Another extension we have introduced is the *probabilistic weight* of arcs. This extension is used when fires of transitions are in conflict or in alternative, and is graphically represented by a numeric value over the arc, in square brackets. Normally, in non-deterministic

situations, which transitions will fire is randomly chosen. With the probabilistic weight, we can instead control this choice, even dynamically.

For example, let us consider the Petri Net in Figure 6 with 3 arcs: A_1 (probabilistic weight $W_1 = 5$), A_2 ($W_2 = 10$), and A_3 ($W_3 = 300$). If at a given time t_1 the choice is between all the three arcs, A_1 shall have a probability of $5/315$ (1.6%) to fire, A_2 a probability of $10/315$ (3.2%), and A_3 a probability of $300/315$ (95.2%). At the time $t_2 > t_1$, let only A_1 and A_2 be enabled: their new probabilities will be $5/15$ (33.3%) and $10/15$ (66.7%) respectively.

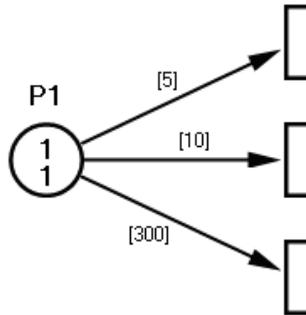


Fig. 6. The probabilistic weight extension

A particular situation occurs when an arc has a probabilistic weight equal to 0. In this case, the associated transition will fire only if there are no other alternative or conflicting arcs with greater probabilistic weight.

2.2 Music Petri nets applicability

At LIM¹, Petri Nets have been applied to music since 1982, and two paths were followed: music analysis and music creation. In other words, in some cases (Degli Antoni & Haus, 1983; Baratè et al., 2005) it has been investigated the possibility of describing causality in music processes through the formal approach of Petri Nets, while other studies focus on music creation ex-novo (Baratè et al., 2007).

It must be noted that different applications of Petri Nets to music analysis lead to apparently contradictory results. While Ravel's *Bolero* structure has been very well described in a convenient series of models (Haus & Rodriguez, 1993), some limitations of this approach have become evident when trying to describe, for example, the complexity of Stravinsky's *Rite of Spring* (De Matteis & Haus, 1996).

From the analytical perspective, excellent or poor results in representing music analysis through Petri Nets formalism mainly depend on three factors:

1. The intrinsic characteristics of the piece to be described. For instance, the music form known as *canon*, based on the literal repetition of the same music objects in different voices at different instants, can be represented in a very efficient and compact way with Petri Nets (see Figure 7).

¹ Laboratorio di Informatica Musicale, Università degli Studi di Milano.

2. The ability in confining those music objects which prove to be efficient from Petri Nets point of view. The concept of music object is deliberately vague and can include whole episodes of a music work, as well as atomic musical events. For this context, *segmentation* will be defined as the activity of isolating music objects and discovering their mutual relationships.
3. The degree of detail the analysis wants to reach. This statement can justify the contradictory results obtained when considering different music pieces. To illustrate this concept, we can consider for instance the test case presented at CMMR 2005 (Baratè et al., 2005), where the first movement of a sonata by W.A. Mozart is modelled. In Figure 8 is presented the very simple and compact net that describes the entire movement at the higher level of abstraction, while in Figure 9 the complexity of the transition in the recapitulation part of the piece is clear even without going into a detailed description of the Petri Net.

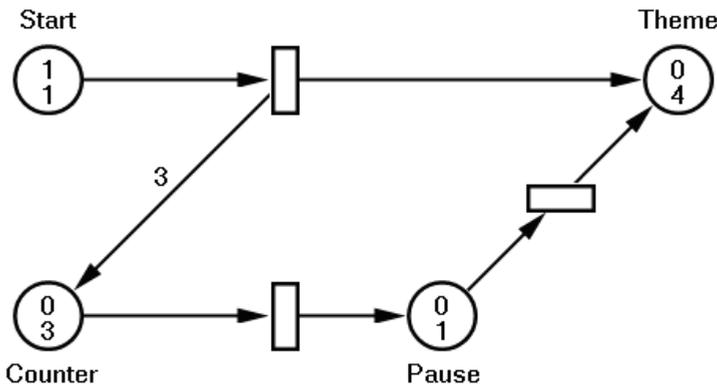


Fig. 7. The Music Petri Net of a canon

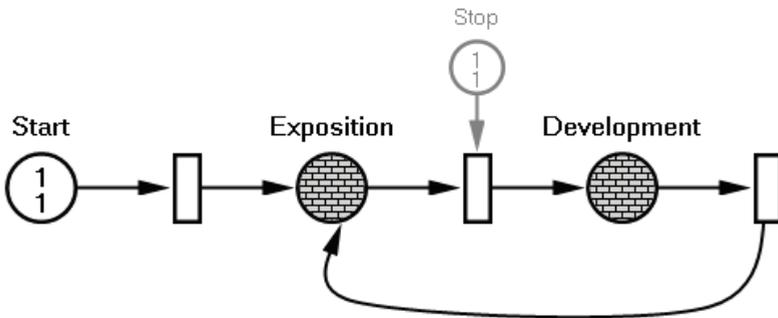


Fig. 8. The Music Petri Net of a sonata form

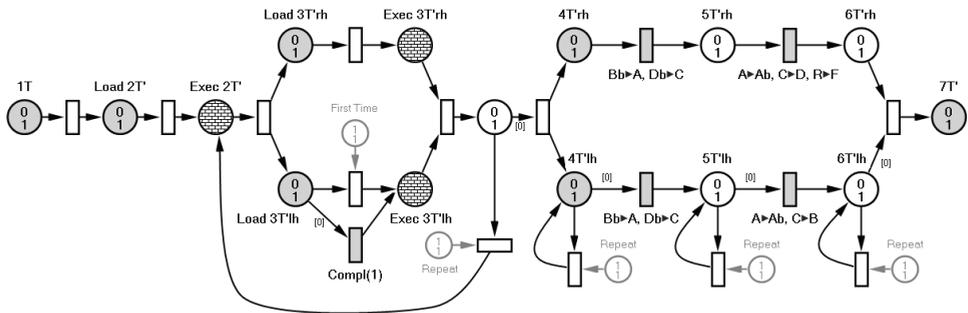


Fig. 9. The Music Petri Net of a part of the recapitulation in a sonata form

2.3. ScoreSynth

Music Petri Nets can be designed, developed and executed with an application named ScoreSynth (Figure 10). This application has an integrated environment to manage complex Petri Nets projects and to execute them in different ways:

- until no transitions are enabled;
- a step every “n” seconds;
- manual step by step;
- manual “object execution” by “object execution”.

In ScoreSynth music objects associated to places are encoded in an XML format named MX.

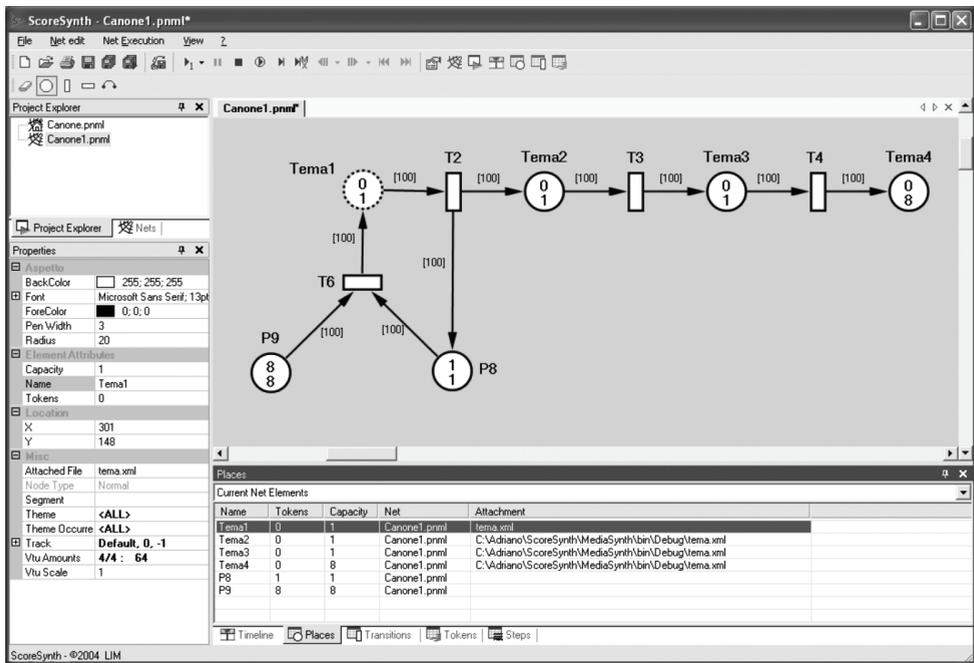


Fig. 10. The ScoreSynth interface

3. MX

MX is an XML dialect currently undergoing the IEEE standardization process (IEEE SA PAR1599). The main purpose of this format is having a comprehensive description of music (Haus & Longari, 2005). Even if specific encoding formats that represent peculiar music features, such as audio tracks or scores, are already commonly accepted and in use, they are not conceived to encode all this features together. On the contrary, we are interested in a comprehensive description of music, containing heterogeneous representations in a synchronized way.

In order to achieve a comprehensive description of music and complete synchronization among both homogeneous and heterogeneous representations of music contents, MX is based on two key concepts: an XML-based multi-layer structure and a space-time construct called spine. In the following sub-sections, we will define these concepts in detail.

3.1 Multi-layer structure

A comprehensive analysis of music richness and complexity highlights six different levels of music description: general, logical, structural, notational, performance, and audio layers (see Figure 11).

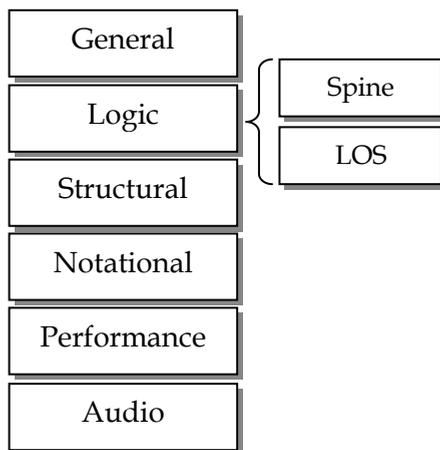


Fig. 11. MX multi-layer structure

General layer is mainly devoted to contain catalogue information about the encoded music piece. Logic layer contains information referenced by all other layers, and it is composed of two elements: the Spine, a sort of a “table of contents” used to mark music events in order to reference them from the other layers and the LOS (Logically Organized Symbols), that describes the score from a symbolic point of view (e.g. chords, rests). Structural layer contains compositional and musicological descriptions of the structure of the music piece (in this layer Music Petri Net links are allowed). Notational layer links visual instances of a music piece, such as digital images of the score. Performance layer links parameters of notes to be played and parameters of sounds to be created by a computer performance (e.g. MIDI and Csound). Finally, Audio layer describes audio information coming from recorded performances.

It must be noted that this approach allows MX to import commonly accepted formats aimed at music encoding, only by linking them in a particular layer, and then creating a mapping of the described events.

3.2 Spine

In order to synchronize the material described in all the MX layers, we introduced the concept of spine, a structure that relates time and spatial information. Spine is made of an ordered list of events, marked through a unique identifier to permit a reference from a particular layer. Each spine event can be described in different layers as well as in different instances within the same layer; e.g., in three different audio clips mapped in Audio layer.

Thanks to spine, MX achieves also a form of synchronization among layers (inter-layer synchronization) and among instances within a layer (intra-layer synchronization). Through such a mapping, it is possible to fix a point in a layer instance (e.g. Notational layer) and jump to the corresponding point in another one (e.g. Audio layer). This peculiarity was used in various applications to allow an evolved and integrated form of music enjoyment (Baggi et al., 2005; Baratè & Ludovico, 2007).

4. Music Petri nets and MX interaction

The adoption of the MX format in order to encode music objects in Music Petri Nets opens new possibilities to analysis and composition of music pieces. In a musicological perspective, an existing piece can be described by Petri Net models that can be linked together with other information in a single MX file. With specific applications, the analyst is able to have a global perspective at various levels of abstraction, described in different MX layers.

Another aspect that takes advantages of this formalism is music creation. By using the MX format a composer could concentrate on the structure of the music piece he wants to obtain, without minding lower level material involved in the mixing process, such as the file formats of the linked objects. Thanks to MX, the final result automatically generates synchronisation of various kinds of music representation, permitting a new type of musical experience.

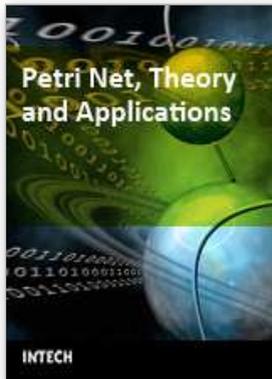
5. Acknowledgments

The author wants to acknowledge researchers and graduate students at LIM, and the members of the IEEE Standards Association WG on MX (PAR1599) for their cooperation and efforts. Special acknowledgments are due to: Denis Baggi, Goffredo Haus and Luca Andrea Ludovico for their invaluable work as working group chair, co-chair, and coordinator of the IEEE Standard Association WG on MX (PAR1599).

6. References

Baggi, D.; Baratè, A.; Haus, G.; Ludovico, L.A. (2006). Developing Intuition in Engineering Education: New Technology To Capture Structures in Music, *Proceedings of the 35th International IGIP Symposium, in cooperation with IEEE / ASEE / SEFI*, Tallinn, Estonia

- Baratè, A.; Haus, G. & Ludovico, L.A. (2005). Music Analysis and Modeling through Petri Nets, In: *Computer Music Modeling and Retrieval*, R. Kronland-Martinet, T. Voïner, S. Ystad (Eds.), pp. 201-218, Springer Berlin Heidelberg, 3-540-34027-0, Berlin
- Baratè, A.; Haus, G. & Ludovico, L.A. (2007). Petri Nets Applicability to Music Analysis and Composition, *Proceedings of the International Computer Music Conference '07 (ICMC 2007)*, Holmen Island, Copenhagen, Denmark, 08-2007
- Baratè, A.; Ludovico, L.A. (2007). An XML-based Synchronization of Audio and Graphical Representations of Music Scores, *Proceedings of the 8th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2007)*, Santorini, Greece
- Degli Antoni, G. & Haus, G. (1983). Music and Causality, *Proceedings of 1982 International Computer Music Conference*, pp. 279-296, La Biennale, Venezia. Computer Music Association Publ., San Francisco
- De Matteis, A. & Haus, G. (1996). Formalisation of Generative Structures within Stravinsky's "Rite of Spring". *Journal of New Music Research*, Vol. 25, No. 1, pp. 47-76
- Haus, G. & Longari, M. (2005). A Multi-Layered Time-Based Music Description Approach based on XML. *Computer Music Journal*. MIT Press
- Haus, G. & Rodriguez, A. (1993). Formal Music Representation; a Case Study: the Model of Ravel's Bolero by Petri Nets, In: *Music Processing*, G. Haus (Ed.), Computer Music and Digital Audio Series, pp. 165-232, A-R Editions, Madison



Petri Net, Theory and Applications

Edited by Vedran Kordic

ISBN 978-3-902613-12-7

Hard cover, 534 pages

Publisher I-Tech Education and Publishing

Published online 01, February, 2008

Published in print edition February, 2008

Although many other models of concurrent and distributed systems have been developed since the introduction in 1964 Petri nets are still an essential model for concurrent systems with respect to both the theory and the applications. The main attraction of Petri nets is the way in which the basic aspects of concurrent systems are captured both conceptually and mathematically. The intuitively appealing graphical notation makes Petri nets the model of choice in many applications. The natural way in which Petri nets allow one to formally capture many of the basic notions and issues of concurrent systems has contributed greatly to the development of a rich theory of concurrent systems based on Petri nets. This book brings together reputable researchers from all over the world in order to provide a comprehensive coverage of advanced and modern topics not yet reflected by other books. The book consists of 23 chapters written by 53 authors from 12 different countries.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Adriano Barate (2008). Music Description and Processing: An Approach Based on Petri Nets and XML, Petri Net, Theory and Applications, Vedran Kordic (Ed.), ISBN: 978-3-902613-12-7, InTech, Available from: http://www.intechopen.com/books/petri_net_theory_and_applications/music_description_and_processing__an_approach_based_on_petri_nets_and_xml

INTECH

open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2008 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.