

Música e o XML

David Freitas (davidfreitas@magicknights.org)
Jorge Amaral (jorge.amaral@netcabo.pt)

12 de Janeiro de 2004



Processamento Estruturado de Documentos

Faculdade de Engenharia da Universidade do Porto

Conteúdo

1	Introdução	1
2	A Música e a sua Representação Electrónica	1
2.1	NIFF (Notation Interchange File Format)	1
2.2	SMDL (Standard Music Description Language)	2
2.3	MIDI (Musical Instrument Digital Interface)	2
3	MusicXML	3
3.1	Algumas Aplicações	3
3.1.1	Migração de música entre software	3
3.1.2	Tradução universal entre métodos de notação musical ocidental	3
3.1.3	Publicação de música em formato não proprietário . .	4
3.1.4	Análise	4
3.1.5	Execução	4
3.1.6	Novas Funcionalidades	5
3.2	Olá Mundo!	5
4	Aplicações	9
4.1	Aplicação Gráfica	10
4.2	MusicXML2SVG	10
4.3	MusicXML2XHTML	11
5	Avaliação do MusicXML	12
A	Noções Essenciais de Teoria Musical	14
A.1	Pauta	14
A.2	Figuras Musicais	15

1 Introdução

A representação de notação musical em formato electrónico não é um fenómeno recente. O MIDI, muito provavelmente o formato musical com maior popularidade, já existe há mais de 30 anos. Estranhamente, para uma área com tanto tempo de existência, até há muito pouco tempo, nenhum formato permitia facilmente ser representado graficamente, reproduzir som e, porque não, ser utilizado via Web. Algumas tentativas de conseguir estes objectivos foram feitas. Poucas tiveram sucesso.

Áreas como a música em formato audio ou livros em formato electrónicos, estão a ser muito explorados enquanto que a publicação de notação musical através da Internet representa um potencial inexplorado. A maioria da música publicada em formato digital está representada em PDF (Portable Document File), que não acrescenta qualquer informação semântica à música representada.

Este artigo pretende analisar o que o XML trouxe de novo a esta área, se este é o bom caminho e, se sim, o que pode ser melhorado. Pretende-se ainda deixar algumas ideias de novas funcionalidades que, com o MusicXML, podem ser mais facilmente implementadas. Durante este estudo utilizaremos como base o MusicXML, formato que começa a ser um Standard entre as principais aplicações musicais.

2 A Música e a sua Representação Electrónica

A representação de música e reprodução é uma operação complexa.

Nesta secção vamos ver alguns formatos que se propuseram a resolver estes problemas: O NIFF (Notation Interchange File Format) e o SMDL (Standard Generalized Markup Language) assim como o formato com maior sucesso: o MIDI.

2.1 NIFF (Notation Interchange File Format)

O NIFF (Notation Interchange File Format) representa música de uma forma gráfica. Não existe o conceito de nota. Todos os elementos: notas, acidentes, tempo, etc. são representados pela sua posição na pauta e/ou pelo seu grafismo. Esse formato é excelente quando a fonte de informação é o scanner e o único objectivo é a representação gráfica. A sua utilização fora das aplicações de digitalização é muito restrita e de uma forma geral mal sucedida. Operações de análise e de reprodução sonora são para além de extremamente complexas, pouco eficientes e passam, geralmente, por uma representação complementar.

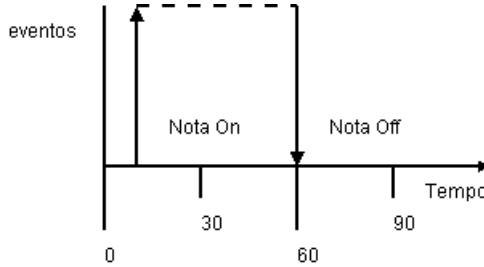


Figura 1: Funcionamento básico do MIDI.

2.2 SMDL (Standard Music Description Language)

O formato SMDL (Standard Music Description Language) foi uma tentativa de criar uma especificação formal para representação musical baseada no Standard Generalized Markup Language (SGML). Foi concebido sem guias de problemas de implementação. Tentou resolver todos os problemas da representação e reprodução do passado, presente e futuro. Obviamente, resultou numa representação extremamente complexa, de tal forma que se diz que esta não é percebida por ninguém. Parece não ter tido quaisquer resultados a nível comercial.

2.3 MIDI (Musical Instrument Digital Interface)

O MIDI - Musical Instrument Digital Interface, é um protocolo de comunicação entre instrumentos musicais electrónicos inventado em 1981 por Dave Smith (co-fundador da empresa Sequential Circuits), que pretendia servir de standard de comunicação, para que as diferentes marcas de instrumentos da época não utilizassem formatos proprietários, implicando que um músico não pudesse trabalhar em simultâneo com instrumentos de diferentes fabricantes.

O MIDI é sem dúvida o único formato que obteve sucesso para representação e reprodução musical em formato electrónico. Neste formato a informação é descrita através de eventos. Cada evento é descrito pelo início (evento on), fim (evento off), e o tempo em que ocorre (definido em unidades de tempo - ticks). Um exemplo pode ser visto na Figura 1.

No entanto, o modo como a informação musical é representada não é adequada para muitas aplicações, como por exemplo, aplicações que realizam análise musical. O formato MIDI normalmente não fornece informações sobre tonalidade, marcação (4/4 ou 2/4) e não distingue notas iguais com nomes diferentes (C# e Db). O MIDI também não é adequado para geração de partituras pois, além das ambiguidades já citadas, a informação rítmica contida num arquivo MIDI é insuficiente para extrair a notação adequada.



Figura 2: Exemplo de um dó em MusicXML.

3 MusicXML

Actualmente, o MusicXML está disponível através de uma licença livre baseada no modelo da W3C e é suportado, entre outros, pelos seguintes programas comerciais: Finale, SharpEye Music Reader e o Dolet.

Existem ainda alguns projectos open source como o XEMO e o KGuitar.

Na realidade, a adopção do MusicXML é a mais rápida desde o MIDI. Só se pode esperar ainda maiores e melhores desenvolvimentos no futuro.

O MusicXML pretende funcionar como um tradutor universal para as notações musicais mais comuns, suportando ainda aplicações de análise, recolha de informação e execução. Existe uma separação dos elementos sonoros dos elementos de notação, permitindo que um arquivo represente uma informação sonora fiel e que a sua representação e execução seja menos limitada. Na figura 2, os nós duration (duração), e type (tipo), apesar de possuírem uma dependência semântica, são definidos separadamente e independentemente.

3.1 Algumas Aplicações

Já são muitas as aplicações que utilizam MusicXML. De seguida descrevemos algumas das funcionalidades já implementadas assim como algumas ideias de possíveis novas aplicações.

3.1.1 Migração de música entre software

Devido ao grande número de formatos proprietários existentes, o simples facto de ser possível, através do MusicXML, permitir a tradução entre estes, é suficiente para causar grande impacto na comunidade de músicos internacional.

3.1.2 Tradução universal entre métodos de notação musical ocidental

Obter notação musical a partir de um ficheiro MusicXML é também uma funcionalidade importante, uma vez que qualquer notação musical pode ser convertida para MusicXML.

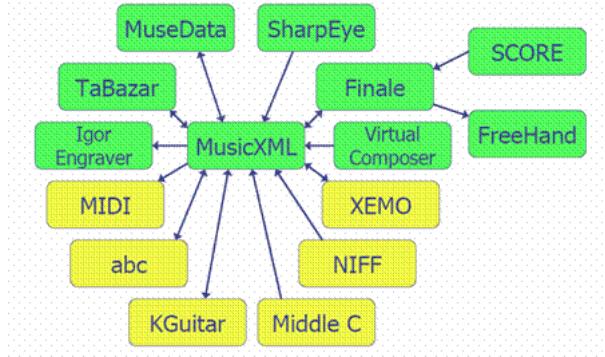


Figura 3: O presente do MusicXML.

Podemos, assim, ter um tradutor de notação universal baseado em MusicXML, sem que se perca informação no processo como acontece com software especializado de notação.

3.1.3 Publicação de música em formato não proprietário

A publicação na Internet, por exemplo, de ficheiros MusicXML, pode resolver os problemas de quem necessita de um formato que permita, não só execução, como também visualização de uma música.

Actualmente, o mais comum é recorrer a serviços de download de ficheiros MIDI e utilizar um conversor MIDI para pauta ou tabulatura com resultados muito imperfeitos. O MusicXML garante a correcta representação visual da música, aliada a uma também correcta execução.

3.1.4 Análise

O MusicXML tira partido de ser um documento XML. Pode utilizar o XQuery, XML Document Object Model (DOM) ou Simple API for XML Parsing (SAX) ou ainda a XML Path Language (XPath) para mais facilmente analisar composições musicais. Deste modo é mais fácil recolher informações de estilo ou fraseamento.

3.1.5 Execução

A incorporação do MIDI no MusicXML permite a comunicação directa com literalmente todos os sistemas electrónicos de execução musical dos últimos 20 anos.

Paralelamente, a execução humana está também garantida através da visualização de pautas e tabulaturas.

3.1.6 Novas Funcionalidades

O MusicXML permite não só melhorar o que já existe, mas também permite, muito facilmente, oferecer novo serviços. Pensamos em duas novas aplicações que poderiam ser úteis, que pensamos ainda não exister:

- Um músico (possivelmente amador) deseja comprar um CD que contenha uma música da qual apenas de lembra da melodia. O nome do artista, música ou album, que hoje em dia são essenciais para comprar o disco não são conhecidos. Ele pode transcrever a melodia utilizando uma aplicação que gere MusicXML e submeter o documento XML, possivelmente a uma empresa, que oferece os seus serviços de consulta à sua base de dados em XML devolvendo-lhe, através da análise da música, o nome do(s) artista(s), música(s) e disco(s), assim como parte(s) da(s) música(s) que a interpretam ou da(s) que mais se parece(m) com a pretendida. Eventualmente poderá através de um serviço web (implementado com WebServices) permitir-lhe a compra on-line do(s) disco(s).
- Uma loja especializada em vendas on-line permite aos seus utilizadores criar um perfil dos seus gostos musicais. Até aqui a sugestão de discos que possam interessar aos cliente é feita pelo estudo dos artistas que gosta, estilo do artista, grupo de utilizadores que têm gostos parecidos, etc. Nunca é utilizado a própria semântica musical: o ritmo da música, os instrumentos, acordes mais ou menos complexos, etc. Com o MusicXML é possível estudar melhor a música que gosta.

3.2 Olá Mundo!

De seguida, apresenta-se um exemplo simples de uma partitura, em MusicXML, relativa à figura 2. Trata-se uma secção de música constituída por um compasso com uma semibreve em dó, baseada num tempo de 4/4.

Uma das representações possíveis para a representar em MusicXML pode ser vista na figura 3.2.

Começando pelo topo do documento, temos:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

Esta é a declaração XML indispensável a todos os documentos XML. Especificamos que os caracteres estarão escritos no codificação Unicode "UTF-8". Esta é a versão do Unicode que inclui o ASCII como subset. Ao definir o valor de standalone como "no" definimos de forma externa o documento a partir de um outro documento.

```
<!DOCTYPE score-partwise PUBLIC  
"-//Recordare//DTD MusicXML 0.8 Partwise//EN"  
"http://www.musicxml.org/dtds/partwise.dtd">
```

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<!DOCTYPE score-partwise PUBLIC
  "-//Recordare//DTD MusicXML 0.8 Partwise//EN"
  "http://www.musicxml.org/dtds/partwise.dtd">
<score-partwise>
  <part-list>
    <score-part id="1">
      <part-name>Music</part-name>
    </score-part>
  </part-list>
  <part id="P1">
    <mesure number="1">
      <attributes>
        <divisions>1</divisions>
        <key>
          <fifths>0</fifths>
        </key>
        <time>
          <beats>4</beats>
          <beat-type>4</beat-type>
        </time>
        <clef>
          <sign>G</sign>
          <line>2</line>
        </clef>
      </attributes>
      <note>
        <pitch>
          <step>C</step>
          <octave>4</octave>
        </pitch>
        <duration>4</duration>
        <type>whole</type>
      </note>
    </mesure>
  </part>
</score-partwise>

```

Figura 4: Representação da figura 2 em MusicXML.

Definimos a utilização do MusicXML, especificamente uma partitura partwise, isto é, onde compassos estão incluídos em partes. Usamos a declaração PUBLIC, incluindo o URL onde se localiza o DTD. Pode-se também usar um URL local como:

```
<!DOCTYPE score-partwise PUBLIC  
"-//Recordare//DTD MusicXML 0.8 Partwise//EN"  
"//C:/MusicXML/dtds/partwise.dtd">  
  
<score-partwise>
```

Este é o tipo raíz do documento. O elemento `<score-partwise>` é constituído por partes, onde cada parte é um agrupamento de compassos. Existe também uma opção `<score-timewise>` que é constituída por compassos, onde cada compasso é um agrupamento de partes.

```
<part-list>  
  <score-part id="1">  
    <part-name>Music</part-name>  
  </score-part>  
</part-list>
```

Tanto usando a definição partwise como a timewise para a partitura, o documento MusicXML começa por listar as diferentes partes da partitura. O exemplo acima é o mais simples possível: contém uma score-part, o atributo obrigatório `id` da score-part e o elemento obrigatório `part-name`.

```
<part id="P1">
```

Temos agora a primeira (e única, neste caso) parte do documento. O atributo `id` de part deve corresponder ao atributo `id` para a score-part referido acima.

```
<measure number="1">
```

Começamos pelo primeiro compasso compasso da primeira parte.

```
<attributes>
```

O elemento `attributes` contém a informação necessária para interpretar as notas e dados musicais desta parte.

```
<divisions>1</divisions>
```

Cada nota em MusicXML tem um elemento duration que define a duração. O elemento divisions define a unidade de medida para o elemento duration em termos de divisões por semínima. Uma vez que neste caso só temos uma semibreve, não necessitamos de dividir a semínima, pelo que definimos o valor de divisions como 1.

```
<key>
    <fifths>0</fifths>
</key>
```

O elemento **key** é usado para representar a escala utilizada, que se reflecte na partitura na indicação de acidentes fixos (ou armadura da clave) junto a clave. Neste caso temos a escala de Dó maior, sem qualquer bemol ou sustenido, o que leva a que o elemento **fifths** seja 0. Se o trecho estivesse em escala de Ré maior, com dois sustenidos, **fifths** seria 2. Se fosse na escala de Fá maior, com um bemol, **fifths** seria -1. O nome "fifths" vem da representação da escala no círculo de quintas. Permite representar as escalas standard apenas com um elemento, em vez de elementos distintos para sustenidos ou bemóis.

```
<time>
    <beats>4</beats>
    <beat-type>4</beat-type>
</time>
```

O elemento **time** representa o compasso da música. Contém dois elementos, **beat** e **beat-type**, que são o numerador e o denominador do compasso respectivamente. Neste caso, 4/4.

```
<clef>
    <sign>G</sign>
    <line>2</line>
</clef>
```

O MusicXML admite vários tipos de claves, incluindo algumas não utilizadas actualmente. Neste caso representa-se a clave de Sol na segunda linha da partitura (a segunda linha a contar do fundo da partitura é um Sol).

```
</attributes>
<note>
```

Aqui terminam os atributos, e começa a primeira nota.

```

<pitch>
    <step>C</step>
    <octave>4</octave>
</pitch>

```

O elemento `pitch` deve conter os elementos `step` e `octave`. Opcionalmente pode ter o elemento `alter`, se houver sustenidos ou bemolis. Estes elementos representam som, de forma que o elemento `alter` deve estar incluído mesmo que a escala esteja definida. Neste caso não temos alterações a nota. O `step` é C (Dó). A definição de `octave` como 4 indica a oitava que começa no Dó central. Assim, esta nota é um Dó central.

```
<duration>4</duration>
```

O valor de `divisions` é 1 por semínima, e como uma semínima corresponde a 4 semibreves, `duration` é 4.

```
<type>whole</type>
```

O elemento `type` indica que a nota é uma semibreve (em inglês "whole note"). Podia-se chegar a esta conclusão pela informação da duração, mas é mais fácil se as informações relativas à notação e à execução musical sejam representadas separadamente.

```
</note>
```

Assim, esta nota está definida.

```
</measure>
```

O compasso está também definido.

```
</part>
```

A parte está também definida.

```
</score-partwise>
```

E finalmente a partitura.

4 Aplicações

Foram desenvolvidas três aplicações com funcionalidades distintas baseadas em MusicXML. Uma aplicação gráfica, construída em Java, que permite visualizar uma pauta, gerar e reproduzir MIDI. As outras duas aplicação permitem converter MusicXML em XHTML e SVG para poderem ser visualizadas directamente num browser.

As aplicações tiveram como objectivo mostrar as potencialidades do MusicXML e não a representação completa da pauta. Estas aplicações são um ponto de partida para uma aplicação mais complexa.

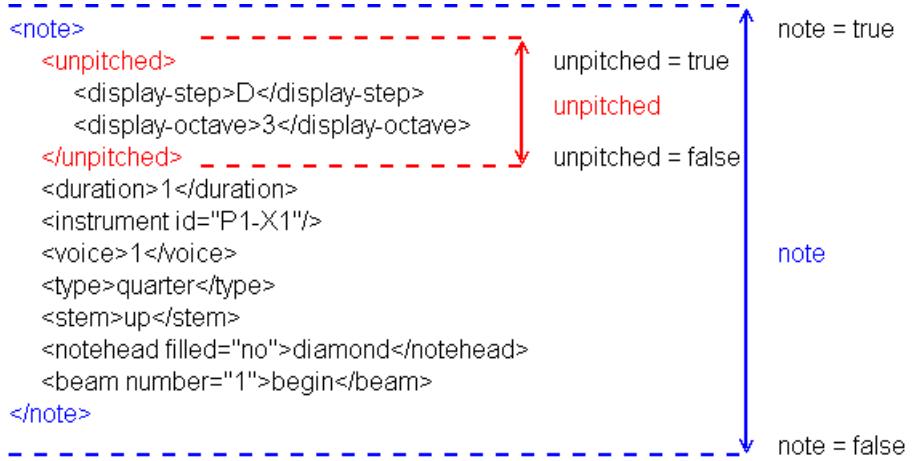


Figura 5: Criação do contexto de cada elemento.

4.1 Aplicação Gráfica

A aplicação gráfica utiliza a API Java - SAX. A sua utilização facilitou imenso o desenvolvimento da aplicação. Esta escolha baseou-se no facto do SAX ser baseados em eventos que era exactamente o que queríamos como explicamos a seguir.

A solução utilizada para enquadrar os vários elementos e seus atributos foi criar um conjunto de variáveis globais (uma por cada elemento) às quais são atribuídos os valores verdadeiro ou falso sempre que se inicia ou finaliza o processamento de determinado elemento. Desta forma é extremamente simples saber em que contexto se encontra o elemento. Um exemplo pode ser encontrado na figura 5.

As notas são desenhadas dinamicamente permitindo assim operações como zoom, mudanças do seu aspecto em tempo-real como por exemplo mudar de cor quando a nota estiver a tocar (ainda não implementado) ou ainda a possível geração de SVG.

A reprodução e sua gravação num ficheiro MIDI foi possível através da API: Java Sound API.

Esta aplicação focou-se especialmente nas pautas para bateria que introduzem diferente notações. Um exemplo da aplicação pode ser visto na figura 6.

4.2 MusicXML2SVG

A aplicação MusicXML2SVG pretende ser uma aplicação simples de linha de comando que permita, a partir de um ficheiro MusicXML obter-se um ficheiro SVG com a visualização da pauta correspondente a este.

Assim, o ficheiro SVG de saída apresenta:

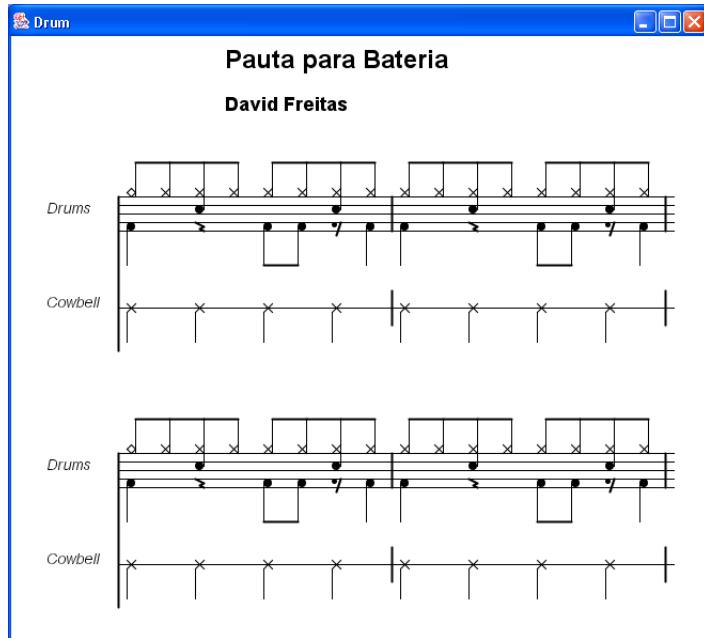


Figura 6: Aplicação gráfica para apresentação de pautas em MusicXML.

- As linhas de pauta necessárias para suportar as notas a representar;
- Claves;
- Tempos;
- Divisão de compassos;
- Notas desde a semibreve à semifusa;
- Pontos de aumento;
- Pausas correspondentes às notas possíveis;
- Apresentação do nome da peça e autores.

Esta aplicação foi desenvolvida em Java, usando SAX para ler o ficheiro XML fazer a construção do ficheiro SVG a partir deste. Trata-se de uma aplicação de consola que recebe o nome do ficheiro XML e o nome do ficheiro de saída desejado.

4.3 MusicXML2XHTML

A aplicação MusicXML2XHTML pretende ser uma aplicação simples de linha de comando que permita, a partir de um ficheiro MusicXML obter-se

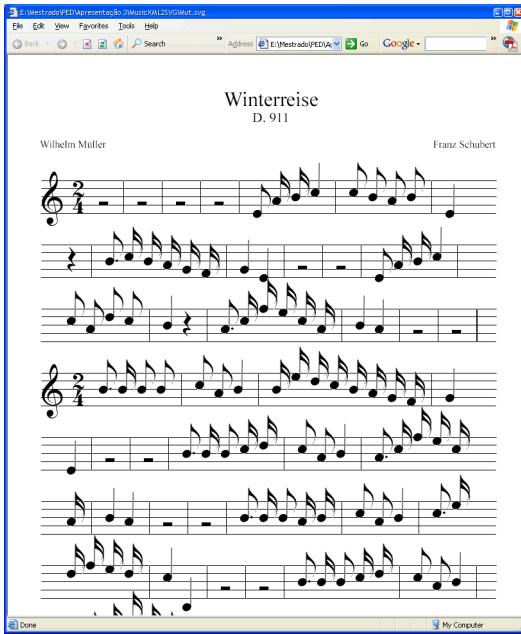


Figura 7: As aplicações MusicXML2XHTML e MusicXML2SVG geram o seguinte resultado

um ficheiro XHTML e um ficheiro CSS que permita a visualização da pauta correspondente a este

Esta aplicação foi desenvolvida em Java, usando SAX para ler o ficheiro XML fazer a construção dos ficheiros XHTML e CSS a partir deste. Trata-se de uma aplicação de consola que recebe o nome do ficheiro XML e o nome base para os ficheiros de saída desejados.

Um exemplo da aplicação pode ser visto na figura 7.

Os resultados desta aplicação são os mesmos que os da aplicação anterior.

5 Avaliação do MusicXML

Apesar de existirem alguns pontos do MusicXML que devem ser melhorados e outros que devem ser estudados, como o facto de ser possível representar pautas incorrectas no MusicXML. Será que não deveríamos impor uma construção correcta da pauta? Pelo menos uma opção para tal. Neste caso a validação não poderia continuar a ser feita por um DTD por não possuir características suficiente e passar a ser validado por um Schema. Uma outra preocupação é a complexidade do MusicXML, que limita muito a possibilidade de escrita directa de MusicXML por um músico, efectivamente levando a que sirva principalmente como forma de armazenar ou transferir músicas. A possibilidade de criação musical directamente em XML é um possível

futuro objectivo que permitiria não recorrer a software de notação profissional para a criação de MusicXML. A definição de um formato reduzido do MusicXML ou de uma nova linguagem para composição musical são duas possibilidades a estudar.

Apesar disto, estamos no caminho certo para a representação de pautas em formato electrónico. Da mesma forma que o MIDI resolveu o problema da compatibilidade e do transporte de música no inicio dos anos 80 dos instrumentos musicais, que não conseguiam que instrumentos de companhias diferentes tocassem em uníssono, levou ao aumento das vendas de instrumentos musicais , o XML (seja com o MusicXML ou outro Standard) pode fazer proliferar a distribuição e negócio de pautas musicais em formato electrónico. Novos serviços, como os que referimos, podem ser implementados mais facilmente e os que existem podem ser melhorados.

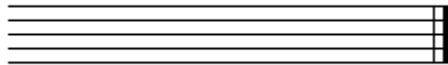


Figura 8: Uma pauta vazia.

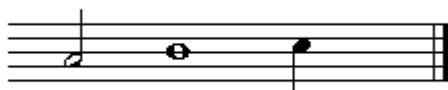


Figura 9: Uma pauta com três notas.

A Noções Essenciais de Teoria Musical

A.1 Pauta

A música tem 3 elementos essenciais:

Melodia combinação sucessiva dos sons;

Harmonia combinação simultânea dos sons;

Ritmo cadência obedecendo a combinação dos sons com periódica repetição.

Usamos a Pauta ou Pentagrama para escrever música (ver figura 8).

As notas musicais podem ser escritas tanto nas linhas como nos espaços da pauta e sabendo assim qual a nota representada. As linhas devem ser contadas de baixo pra cima e o primeiro espaço encontra-se entre a primeira e segunda linha (ver figura 9).

A **Clave** é um sinal colocado no início da pauta e dá nome à nota que está na mesma linha dela.

É importante observar o tipo de clave para poder ler as notas.

Há vários tipos, porém as mais comuns são a Clave de Sol (ver figura 10) que é usada por instrumentos de alto alcance sonoro como o violino, flauta, trompete e clarinete.

E a Clave de Fá que é usada por instrumentos de baixo alcance sonoro como o baixo e violoncelo (ver figura 11).

Antes de saber como se escreve as notas na pauta, vamos ver a ordem e seus nomes.

O nosso sistema de música tem 7 notas. A ordem é Dó, Ré, Mi, Fá, Sol, Lá e Si. Ou em cifras respectivamente C, D, E, F, G, A, e B. Correspondem às teclas brancas (naturais) do piano com pode ver na figura 12.

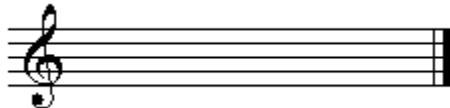


Figura 10: A clave de sol.

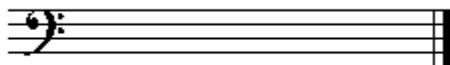


Figura 11: A clave de fá.

A maior distância entre duas notas é chamada de Tom e sua metade de meio tom ou semitom (ver figura 13).

No exemplo da figura 14, podemos ver as notas representadas por cada uma das linhas e espaços da pauta com a Clave de Sol. A primeira linha corresponde à nota Mi (E) e o primeiro espaço à nota Fá (F). Em outras palavras, as notas na pauta (linha-espaço-linha...) seguem a ordem natural (Dó, Ré, Mi, Fá, Sol, Lá e Si). Pode-se também escrever sob a primeira linha e acima da quinta.

Além dos espaços e linhas da pauta, podemos também adicionar linhas extras entituladas Linhas Suplementares Superiores ou Inferiores para poder escrever notas mais agudas ou graves. Um exemplo pode ser visto na figura 15.

A.2 Figuras Musicais

As Figuras Musicais ou Figuras Positivas permitem-nos especificar a duração do som.

Dos tipos de figuras musicais, a breve (vale o dobro da semibreve) e a quartifusa (vale metade da semifusa e tem um colchete a mais) são encon-

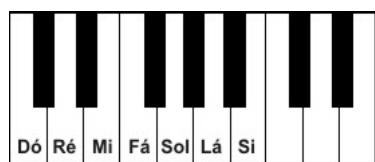


Figura 12: O Nossa Sistema de Música.

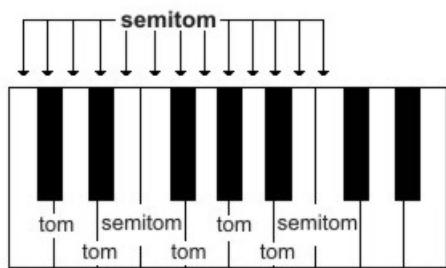


Figura 13: Distância Entre Notas.



Figura 14: Exemplo na Clave de Sol.



Figura 15: Exemplo de Linhas Suplementares.

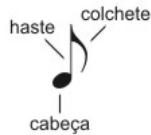


Figura 16: Descrição das Partes de Uma Nota.

Nota	Nome	Valor
●	Semibreve	1
○	Mínima	2
■	Semínima	4
♪	Colcheia	8
♫	Semicolcheia	16
❀	Fusa	32
❀❀	Semifusa	64

Figura 17: Descrição dos Tempos de Uma Nota.

tradas em músicas muito antigas, porém já não são mais usadas actualmente (ver figura 16).

O valor depende de qual Compresso é uma determinada música. Mas tomando como base a semibreve, temos os seguintes números. Exemplo: 1 semibreve = 2 mínimas = 4 semínimas...etc (ver figura 17).

Quando escrevemos colcheias, semicolcheias, fusas e semifusas, costuma-se agrupá-las usando-se linhas (proporcionais aos colchetes da nota) para facilitar a leitura. Um exemplo pode ser visto na figura 18.

Adicionando um ponto ao lado da nota, aumentamos em metade o seu valor (Ver figura 19). Um segundo ponto aumentará em metade o valor do primeiro ponto, e assim por diante.

Podemos ligar notas adicionando uma linha curva que recebe o nome de Ligadura, que indica que não deve haver interrupção. No exemplo da figura 20, vemos dois Compassos iguais, diferenciando apenas a forma escrita.

Existem ligadura de:

valor união de notas de mesmo valor e altura (ver figura 21);



Figura 18: Exemplo de um Agrupamento de Notas.

SEM ponto		COM ponto	
Nota	Equivalente à	Nota	Equivalente à
o	dd	o.	dd d
d	dd	d.	dd d
d	ññ	d.	ñññ

Figura 19: Utilização do Ponto de Aumento.



Figura 20: Utilização da Ligadura.



Figura 21: Ligadura de Tipo Valor.



Figura 22: Ligadura de Tipo Portamento.

Portamento ligação de notas de alturas diferentes (ver figura 22);

frase notas de diferentes compassos ligadas (ver figura 23).



Figura 23: Ligadura de tipo Frase.