Open Score Format

PVG Digital Score Application Profile Specification

Version 1.0

Date: 15/9/2009

Revision History

Date	Version	Description	Author
21/10/08	0.9	Public beta	Mark Olleson, Yamaha R&D Centre London. Michael Good, Recordare LLC
14/9/09	1.0	Open Score Format 1.0 Release	Mark Olleson, Yamaha R&D Centre London.

Acknowledgements

The following organizations have contributed to the development of the Open Score Format

Table of Contents

Open Score Format - PVG Digital Score Application Profile Specification

1. Introduction

1.1 Purpose

The *Open Score Format PVG Digital Score Application Profile* specifies the subset of MusicXML score elements that are required to represent the score features of commercial Piano, Voice and Guitar (PVG) scores – the most commercially significant digital score content genre.

The motivation for specifying a subset of MusicXML is reduce the complexity of application software that uses these scores by eliminating MusicXML features that are not required.

1.2 Scope

This document describes the *Open Score Format PVG Digital Score Application Profile* primarily in terns of the way in which it differs for MusicXML 2.0 Strict; as is the case with MusicXML 2.0 Strict, the formal definition is provided by a W3C XML Schema.

A section detailing certain aspects of MusicXML that are not clear from the schemas or accompanying documents is also provided for the benefit of developers.

Scores conforming to *Open Score Format PVG Digital Score Application Profile* are distributed in Open Score Format Packages alongside metadata and optional media objects. Refer to the *Open Score Format Packaging Specification [1]* for further details

1.3 Intended audience

This document is primarily intended for developers of music application software that creates, manipulates, displays or sells digital score content.

1.4 Definitions, Acronyms and Abbreviations

PVG: Piano Voice and Guitar –a genre of score. PVG scores contain a piano grand-staff, a vocal staff and guitar chord-frames. In terms of value of sales, PVG scores are the most commercially significant content form.

1.5 Conventions

Fragments of XML and file-paths appear in a blue mono-spaced type-face:

- XML fragment: `<dc:title>`
- File-path: `META-INF/Manifest.xml`

1.6 References

[1] Open Score Format Packaging Specification – http://openscoreformat.sourceforge.net/assets/PackagingSpecification.pdf

[2] Open Score Format PVG Application Profile Schema - http://www.musicxml.org/xsd/osfpvg.xsd

[3] MusicXML 2.0 Strict Schema - http://www.musicxml.org/xsd/musicxml.xsd

1.7 Overview

The Open Score Format PVG Application Profile formally specified by a W3C XML Schema [2]. This document provides:

- A description of the differences between the OSF PVG Application Profile and the MusicXML 2.0 Strict [3] schema
- A set of implementation notes

2. Introduction

2.1 The Open Score Format

The Open Score Format is an open file format for the distribution and archival of digital scores.

The Open Score Format builds upon MusicXML 2.0, using it as the basis for the representation of the score components of an OSF package.

In order to reduce the complexity of systems that read, write, render or otherwise process score data, the Open Score Format introduces the concept of ***profiles***: a subset of score features that are required for particular types of content.

The first release of the Open Score Format introduces a profile for the PVG (Piano, Voice, Guitar) content type. Profiles specific to other content types may follow.

2.2 PVG Content

PVG (Piano, Voice, Guitar) content is the most commercially significant content form sold today by music publishers. For this reason it was chosen as the first content profile for the Open Score Format.

PVG Content is predominantly used for popular songs. It comprises a Grand Staff containing the piano (or keyboard) part and a vocal staff containing the lead vocal line and lyrics. The guitar part is notated using chord frames over the system. In some scores, additional lyrics for repeat sections are printed after the end of the score.

PVG Content specifically does not include guitar or bass tablature parts.

2.3 Differences between MusicXML 2.0 and OSF PVG Application Profile

The Open Score Format PVG Application Profile is primarily comprised of a W3C XML Schema derived by restriction and extension from a strict W3C XML schema version of the MusicXML 2.0 format.

The MusicXML 2.0 format is defined with an XML DTD. The W3C XML schema version of the MusicXML 2.0 specification enforces restrictions that are documented in the MusicXML 2.0 DTD. Since it validates fewer documents than the original DTD, it is called the MusicXML 2.0 Strict schema. The *Open Score Format PVG Application profile* restricts the MusicXML2.0 strict schema even further.

The subset of MusicXML features required by PVG scores has been determined by a review of representative content.

The Open Score Format PVG Profile differs from the MusicXML 2.0 Strict schema in the following ways:

- Features that do not occur in content that meets the strict definition of PVG are removed.
- Features in MusicXML 2.0 that are deprecated but remain extant to maintain backwards compatibility are removed.
- Previously optional attributes and elements (particularly those associated with positional data) are now mandatory.
- Feature definition restrictions that were not clear in the DTD and thus were partially restricted in the Strict schema are now fully restricted in the profile. This mainly affects whether numbers can be integer or decimal values.

- Features that are missing from MusicXML 2.0 but are needed for fully accurate formatting of PVG content have been added (e.g. explicit indication of muted vs. unplayed strings in chord diagrams).

These changes are discussed in detail in section 4 of this document.

3. Implementation Notes

This section provides guidance to developers on certain aspects of MusicXML that are not immediately clear from the documentation that accompanies the schema.

We identified these ambiguities through monitoring MusicXML common development questions on the mailing list and evaluating our personal implementation experience.

3.1 Decoding Voices and Backup

Backup is the mechanism by which multiple voices in a measure are described in MusicXML.

In piano music, the use of multiple voices is extremely common, occurring in even the simplest content examples.

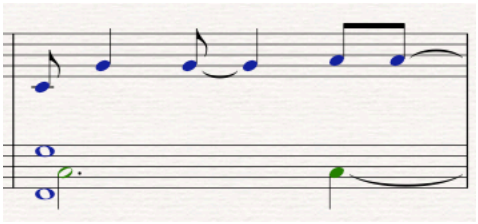

Figure 1: Multi-voice music. Blue note-heads are voice 1 and green voice 2.

In a MusicXML document, notes, Chords and rests from all voices in a measure appear as children of a `measure` element.

The rules for allocating Notes, chords and rests to voices is as follows:

- Notes appear in time-order and are consecutive in terms of timing (e.g. the second note starts at the instant that the previous note ends). Rests are note elements without a child `pitch` element. Each `note` element contains a `voice` element indicating the logical voice in which the note belongs. In a multi-stave instrument such as the Piano, a voice may span both staves.

  A good way to conceptualise this is by thinking of a beat-cursor that indicates the start beat of the next note. When a `note` is read from the measure, the cursor advances by the note's duration. A single beat-cursor is used for all voices.
- A `backup` element is used to move the beat-cursor backwards. This is often used to introduce a new voice, although a `backup` element can apply to the same voice too.
- A `forward` element is used to move the beat-cursor forwards.
- A `note` that has a `chord` child element belongs to a chord.
  - One note of the chord does not have a `chord` child element. This is the *reference note* of the chord.
  - The chord's start beat is the same as the reference note.
  - The beat-cursor does not advance.
  - The duration of the note must be the same as the reference note. Chords whose notes have different lengths are not supported[1]

[1] Use of additional voices can be used to notate scores in which this is desired.

- Use of backup or forward between notes in a chord results in undefined behaviour, and will fail validation with the OSF validation tool.

- Voices may cross staff boundaries. This is common of piano music. Encoding cross-staff passages as a single voice is desirable as extra musical semantics are captured (e.g. a complete phrase).
- A particular voice need not necessarily contain notes or rests across every beat of the measure. However, when taken as a whole, all voices in the measure will meet this condition.
    - Don't assume that notes are consecutive or that the voice will have notes at either the beginning or end of the measure

This process is described as *voice analysis* in the following sections.

3.2 Handling Ties

Determining the correct location of the start and end of ties has some complicated aspects:

- Ties can span an arbitrary number of measures.

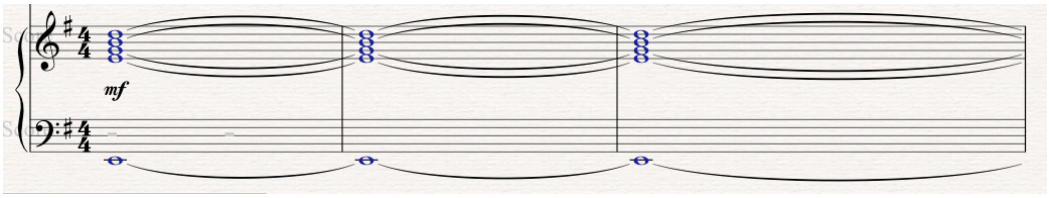

Figure 2: Ties spanning multiple measures

- There are legitimate cases where a tie may have no start or end – for instance in score excerpts where the first or last note of the tie has not been included, or around repeat markings. This is most commonly seen in first and second-time repeat measures where a note is tied from the measure to one of the first- or second-time measures but not the other.

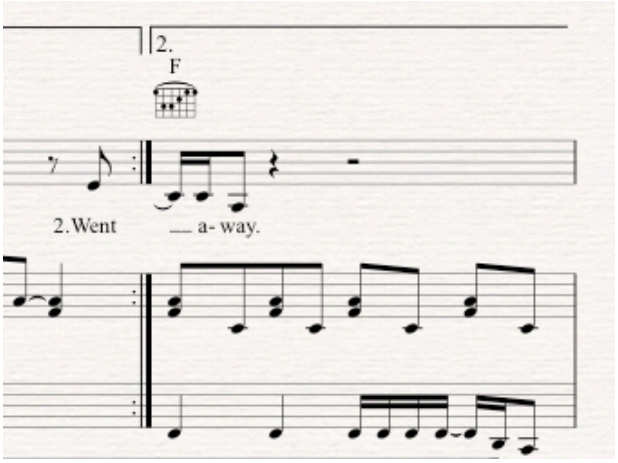

Figure 3: Tie without start (tied from measure before 1st time repeat)

- Notes covered by a tie that are neither the start nor end are not specially marked up.
- Output from some tools may include notes tied across voices. This can complicate tie-analysis, and also result in ties whose end-point is ambiguous.

3.2.1 Tie Analysis

Tie analysis is the process of determining the start and end-points of ties. Voice analysis is required before Tie analysis can take place.

The process of tie analysis is much easier if data structures allow easy linear traversal of notes in a voice across measure boundaries.

A note is eligible to be tied to another:

- Whose **pitch** element has the same **step** and **octave**, and **alter**[2] values.
- When it starts at the same instant as the end of the previous note, with no intervening rest or gap in the voice.

When a tied note is encountered, candidates for the next note in the tie should be considered in this order:

- If the note's end is not at the end of a measure, the next, eligible note in the same measure
- If the note's end is at the end of the measure
    - A note that begins at the beginning of the same voice in the next measure.
    - A note that begins at the beginning of a different voice in the next measure.

The tie ends if the note contains **<tie type="stop">** child element.

[2] The pitch of the note is absolute and does not take into account the notation rule for carrying accidentals across measures in a tie.

4. Differences between MusicXML 2.0 and OSF PVG Profile

4.1 Non-PVG features removed

The Open Score Format PVG profile removes many features that are not used in the PVG repertoire. This simplifies the development and testing of applications that make use of digital scores.

These removed features are:

- **Non-PVG clefs.** Only the G and F clef signs are supported. The `C`, `percussion`, `TAB`, and `none` values for the `sign` element are not supported.
- **Transpositions.** The `transpose` element is removed.
- **Tab features.** The `staff-tuning` element, `capo` element, and `show-frets` attribute are removed from the `staff-details` element.
- **Figured bass.** The `figured-bass` element is removed.
- **Symbols over barlines.** The `fermata`, `coda`, `segno`, and `wavy-line` elements are removed from the `barline` element.
- **Non-PVG noteheads.** The `shape-note` notehead element values (`do`, `re`, `mi`, `fa`, `so`, `la`, `ti`) are removed, as are the `square`, `cross`, `arrow down`, `arrow up`, `slashed`, and back slashed `notehead` element values.
- **Non-PVG dynamics.** The `rf` and `rfz` elements are removed from the dynamics element.
- **Non-PVG directions.** The `accordion-registration`, `damp`, `damp-all`, `eyeglasses`, `harp-pedals`, and `scordatura` elements are removed from the direction-type element.
- **Non-PVG ornaments.** The `shake` and `schleifer` elements are removed from the `ornaments` element.
- **Non-PVG technical indications.** All `technical` elements except for `fingering` and `bend` have been removed. This includes the `up-bow`, `down-bow`, `harmonic`, `open-string`, `thumb-position`, `pluck`, `double-tongue`, `triple-tongue`, `stopped`, `snap-pizzicato`, `fret`, `string`, `hammer-on`, `pull-off`, `tap`, `heel`, `toe`, and `fingernails` elements.
- **Non-PVG barline styles.** The `heavy`, `short`, and `tick` values for the `bar-style` element are removed.
- **Multimetric music.** The `non-controlling` attribute of the measure element is removed. This feature is used to represent measures where the left barlines do not coincide between parts in a score.
- **Modes.** All `mode` element values except `major` and `minor` are removed.
- **Quarter-tone accidentals.** The `quarter-sharp`, `quarter-flat`, `three-quarters-sharp`, and `three-quarters-flat` values are removed from the accidental element.
- **Per-staff key and time signatures.** The `number` attribute of the `key` and `time` elements are removed. These attributes allowed different key and time signatures to appear in separate staves within a single (usually keyboard) part.
- **Detailed key signature appearance.** The `key-octave` element is removed. This element specifies which octave to use for each accidental within a key signature.

- **Single-number time signatures.** The `single-number` attribute is removed from the time element.
- **Functional harmony.** The `Neapolitan`, `French`, `Italian`, `German`, and `Tristan` values for the kind element are removed. The function and inversion elements are removed from the harmony element. MusicXML 2.0 provides these as alternatives to the chord symbol notation used in PVG repertoire.
- **Double and triple underlines.** The values of 2 and 3 are removed from the `number-of-lines` attribute.
- **Fanned beams.** The `fan` attribute for the `beam` element is removed.
- **Extension elements.** The `other-dynamics`, `other-articulation, other-ornament`, and `other-technical` elements are removed.

4.2 Deprecated features removed

The *Open Score Format PVG Application Profile* removes features that were deprecated in MusicXML 2.0 but retained for backward compatibility. These removed features are:

- All attributes of the `part-name` and `part-abbreviations` elements except for the `print-object` attribute. The `part-name-display` and part-abbreviation-display elements should be used instead.
- All attributes of the `group-name and group-abbreviation` elements. The `group-name-display` and `group-abbreviation-display` elements should be used instead.
- The `staff-spacing` attribute of the `print` element. The `staff-layout` element should be used instead.
- The `pan` and `elevation` attributes of the `sound` element. The `pan` and `elevation` elements should be used instead.
- The `directive` element. The `directive` attribute should be used instead.
- While not labelled as deprecated, the `repeater` attribute for beams has largely been superseded by the `tremolo` element, and has been removed.

4.3 Optional features made mandatory

The Open Score Format PVG profile makes certain features mandatory that are optional in the standard MusicXML definition. These now mandatory features are:

- **Full voice specification.** The `voice` element is required within the `note` and `forward` elements.
- **Exact horizontal position** for notes and other musical elements using the `default-x` attribute. The use of the `relative-x` attribute is prohibited.
- **Exact vertical positions** for musical elements other than notes using the `default-y` attribute. The use of the `relative-y` attribute is prohibited.
- **Exact curvature data** for slurs and ties using the `bezier-y` attribute and either the `bezier-x` or `bezier-offset` attributes.

Unfortunately, the current status of nearly all MusicXML software, including the Dolet plug-ins for Finale and Sibelius, make it infeasible to enforce the formatting-related restrictions. To do so would make nearly all MusicXML files invalid according to the PVG schema. Thus the current version of the PVG schema does not enforce the last three restrictions listed here. The schema has been defined to make it simple to add these mandatory features once software technology makes it practical to do so.

4.4 Flexible features now fully restricted

The Open Score Format PVG profile restricts several MusicXML features that were more flexibly – or somewhat unclearly – described in the MusicXML 2.0 definition. These restricted features are:

- Values for the **divisions** element are restricted to positive integers, rather than positive numbers.
- Values for the **type** attribute of the **line-width** element are restricted to the standard choices listed in the MusicXML definition, rather than being any valid XML attribute value.

4.5 PVG features added

The Open Score Format PVG profile adds some features that are required for archival representation of PVG repertoire. These new features are:

- **Explicit muted indication in chord diagrams.** The **fret** element now has a muted attribute to explicitly indicate if a muted x symbol appears in the chord diagram for a given string.
- **Hyphen distance.** Open Score Format PVG files can now specify a standard distance between lyric hyphens for use on a score-wide basis.
- **Word extension formatting.** The extend element now has optional type and position (**default-x, default-y**) attributes to allow more detailed specification of word extension formatting. If not specified, the end of word extensions takes place at the right edge of a note, not the left edge that is standard for other MusicXML elements.
- **Dashed line formatting.** Attribute for **dash-length** and **space-length** are added to elements that can represent dashed lines. These are the **bracket**, **dashed**, **doit**, **falloff, glissando, octave-shift**, **plop**, **scoop**, **slide**, **slur**, and **tied** elements.

5. Schema Documentation

Automatically generated documentation for the OSF PVG Schema can be found at the Open Score Format project website:

http://openscoreformat.sourceforge.net/SchemaDoc/osfpvg1.0.html

Copyright © 2008 Yamaha Corporation, All rights reserved Page 16 of 16