XML Protocol for the Makimono Project

Philippe Macnab-Séguin and Dominique Lafortune

April 2019

The following paper describes the XML search and creation protocols as of April 2019, detailing both the work that has been completed to date as well as tasks to be completed in the near future (End-of-May 2019).

XML file Criteria

XML files being currently created for use in the Makimono project conform to the following criteria:

1. The XML file must be easily **searchable**
2. The XML file must be **graphically professional** when visualized in OrchView/OrchPlay
3. For functionality in OrchView/OrchPlay, a **total of four versions** are needed: Separate/Transposed **(ST)**, Separate/Concert Pitch **(SC)**, Combined/Transposed **(CT)**, Combined/Concert Pitch **(CC)**
    - Separate: Each instrument on its own separate staff, to make annotation and search simpler
    - Combined: Instruments on combined staves (for example, Fl. 1.2.) to improve readability

Parts A & B of this document outline how these criteria have been met through the development and use of protocols and programming tools in the 2018-2019 academic year.

PART A – SEARCH PROTOCOLS

The entire, detailed, search protocol information can be found in the Excel document:
<DATA in OrchSim files and xml transfer 2019-04-19.xlsm>

1. Metadata

See <Metadata> sheet in <DATA in OrchSim files and xml transfer 2019-04-19.xlsm>

Syntax:
Composer_Piece Title_Catalog No._Movement_Date_Country of Origin_Orchestrator

The <Metadata> sheet in our Excel document contains alternate spellings of each composer, and notes on exactly how to write in each field of information so that it remains consistent.

2. Instrument Naming

See <Instrument names> sheet in <DATA in OrchSim files and xml transfer 2019-04-19.xlsm>

We now have an up-to-date list of exact instrument names that will be used throughout the project and their abbreviations, along with protocols for combining multiple instruments on a single staff. Ex.:

- Flute 1 and Flute 2 on a single staff = Flutes 1.2.
- Bass Trombone and Tuba = Bass Trombone + Tuba
- Flute 1 doubling on piccolo = Flute 1 [+picc.]
- Protocol for naming 'grand staff' instruments
  Etc…

3. Playing Techniques Dictionary

See <Playing Techniques dictionary> sheet in <DATA in OrchSim files and xml transfer 2019-04-19.xlsm>

The dictionary contains all playing techniques used so far in the XML files of the OrchPlay library along with their equivalencies. *(Ex.: or, ord., norm. and senza sord. must all be considered equivalent to the playing technique "ordinario" in our search functions)*

4. Percussion Mapping

See <OrchSim Percussion mappings> sheet in <DATA in OrchSim files and xml transfer 2019-04-19.xlsm>

The mapping contains all of the following information for each midi note/percussion mapping combination used in OrchSim (for a total of **450 different combinations**):

- Section
- Instrument
- Instrument Specifics (size, with/without snares, etc.)
- Playing Technique
- Mallet Type

PART B – GRAPHIC REPRESENTATION

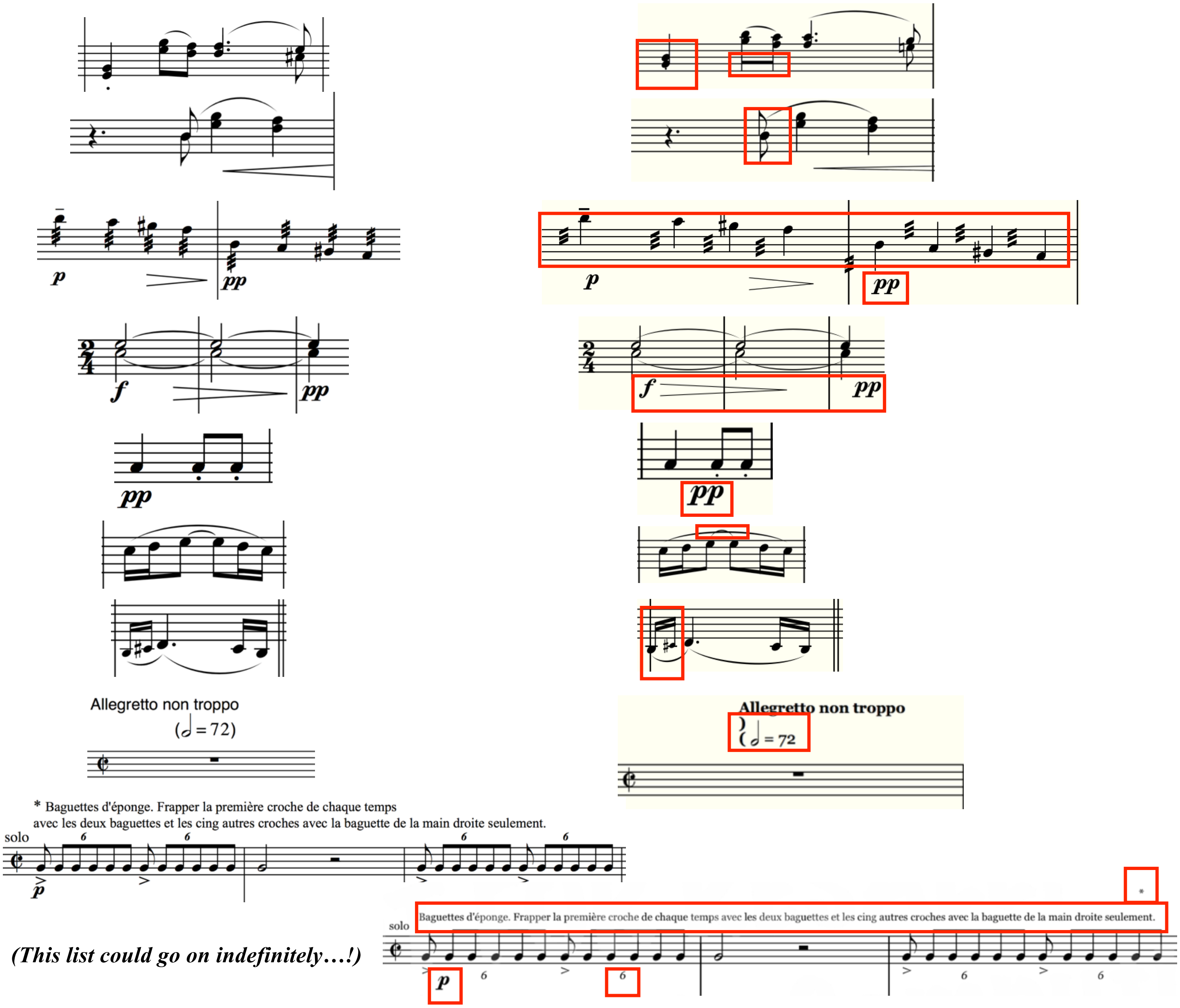
5. The Problem of Graphic Representation

When exporting an XML file from Finale, Sibelius, or any other notation software and opening it in a another program, many details of the score may have changed due to the various ways each of these programs export or read XML data. Slurs may not look the same in each program, each program may have a different way of attaching hairpins to notes, and the alignment of expressions may change. This means that very specific protocols and Python scripts must be created from scratch to ensure that the XML files we export from Finale will display correctly in SeeScore (the program we are using to graphically represent the XML files).

Here are some examples of consistent errors that occur when exporting Finale files from the OrchPlay library as XMLs and opening them directly in SeeScore:

Finale:

OrchView/OrchPlay (SeeScore)
Without editing the XML/Finale file:


p
pp
f
pp
pp
Allegretto non troppo
(𝅗𝅥 = 72)
* Baguettes d'éponge. Frapper la première croche de chaque temps
avec les deux baguettes et les cinq autres croches avec la baguette de la main droite seulement.
solo
6
p
Allegretto non troppo
(𝅗𝅥 = 72)
*
Baguettes d'éponge. Frapper la première croche de chaque temps avec les deux baguettes et les cinq autres croches avec la baguette de la main droite seulement.
solo
6

(This list could go on indefinitely…!)

6. **Python Script** (Philippe)

In order to save time, any recurring errors we have found in the XML export process have been automated with the use of a Python script. Before opening the XML in OrchView/ OrchPlay, we run it through this Python script that alters the XML in various ways to make it readable by SeeScore.

The following is a brief overview of what the script currently does, with further developments being made on a continual basis as we encounter new problems:

a) Fixes "shorthand rests"

Some rests are exported in a "shorthand version", which SeeScore often reads incorrectly. The script places these rests in the proper position based on what clef is currently in use.

b) Fixes displaced dynamics

The placement of dynamics can be indicated in two ways in XML files: "default spacing" (spacing in relation to the bar) or "relative spacing". All of the dynamics that used "default spacing" were incorrectly positioned, so a script had to be made to convert these to "relative spacing", in addition to offsetting each expression over slightly to the left.

c) Fixes tremolos

SeeScore reads tremolos correctly when they had been inputted into finale as expressions, but not as articulations.

d) Fixes fermatas

Due to differences in the way Finale and SeeScore align objects, all fermatas were too far to the left.

e) Makes slurs slightly rounder

The slurs in SeeScore are all slightly less round than those in finale – sometimes this causes collisions, so I made a script that makes all slurs slightly rounder.

f) Fixes 8va problem

Because of a difference in the way Finale and SeeScore include/exclude the final note below an 8va sign, the first note following all 8va signs was placed an octave too high. For example:

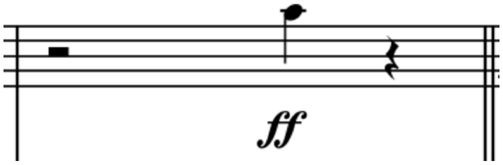
g) Readjusts dynamics that have been offset by hairpins (currently in the making)

Dynamics that appear after the end of a hairpin (crescendo/decrescendo) in the same measure, in any bar, are displaced slightly to the right. Most of the script is complete, and I am currently testing it in various situations.

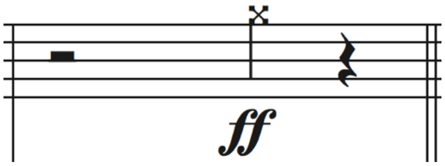
h) Percussion display script (currently in the making)

Finale displays percussion graphically with the use of *staff styles* – however, no staff style information is exported into XMLs. This means that the mapping used for search will also be used to convert the percussion notes in the XML to the correct staff placement/notehead.

Without Percussion Mapping (incorrect):



With Percussion Mapping (correct):



7. Finale Export Protocol

In order to arrive at an XML file that is ready to be run through the Python script and read in OrchView/OrchPlay, a detailed protocol is used. The following is a simplified version intended only to give a brief overview of the process. Approximate time taken for each step of the process has been indicated, with the "max time" indicating how long it can take when a document is long, complex, and poses many new challenges that must be tackled for the first time.

A detailed version of this protocol can be found in <XML Dom-Phil/XML Protocol Template 2019-04-19>

Step 1: creation of "prep" document

In this step, a preliminary document is created that contains both the "combined" and "separate" staff- versions, and has the best layout possible for this stage in the process. It can take anywhere between 2h and 18h.

Dominique – from 0.5h to 5h, currently ca. 1h:

a) Enter Metadata
b) Rename staves
c) Delete unnecessary staves
d) Make sure that instruments are assigned to the same staves consistently throughout the piece *(often a very deceptively complex procedure!)*
e) Delete all playback data *except* tempi, including manually removing hidden and half-hidden expressions
f) Add percussion mappings as invisible text
g) Readjust if the final page is a half page
h) Readjust layout, especially for "2nd parts" (Flute 2, etc. which didn't appear in original layout)
i) Fix 8va attachments
j) Combine staves if they are not already combined and correct/adjust layout accordingly
k) Re-adjust 2nds between layers (SeeScore adds space between 2nds that must be compensated for)
l) Make hairpins attach to beats properly

Philippe – from 0.5h to 8h, currently ca. 2h:

m) Export Finale file as XML
n) Run XML through Python script and open in OrchView or SeeScore
o) Check entire XML document to find errors, edit Finale file or Python script, and go back to m)

Dominique – from 0.5h to 1.5h, currently ca. 1h:

p) Check entire XML document in OrchView or SeeScore, make a list of errors

Philippe – from 0.5h to 3h, currently ca. 1h:

q) Correct the document and do a final triple-check

Step 2: Creation of four XML files for use in OrchView/OrchPlay

In this step, the prep document is used as a basis to make the ST, SC, CT, CC XML files. It can take from 3h to 9h.

Dominique – from 1h to 3h, currently still learning/speeding up the process:

a) Delete 2nd Layer for all combined instruments, rename staves, and adjust transposing instruments to make "Separate/Transposed" (ST) score
b) Remove transpositions from ST score and adjust layout to create "Separate/Concert Pitch" score (SC)
c) Delete extra instruments from prep document (Flute 2, Clarinet 2, etc.) and adjust for transposing instruments to make "Combined/Transposed" (CT) score
d) Remove transpositions from CT score and adjust layout to create "Combined/Concert Pitch" score

Philippe – from 0.5h to 2h, currently ca. 1h:

e) Export the four Finales file as XMLs
f) Run all four files through Python script and open in OrchView or SeeScore
g) Check all four XML documents to find errors, edit Finale files or Python script, and go back to e)
Dominique – from 1h to 2h:
h) Check all four XML documents in OrchView or SeeScore, make a list of errors
Philippe – from 0.5h to 2h:
i) Correct the documents and do a final triple-check

We have found that the recent addition of Dominique Lafortune to this process has been essential in reducing errors and increasing efficiency. Every error that is caught in the "prep" stage is one that we would have had to correct four times if not caught before Step 2.

To give an idea of the number of errors that can occur in single a document, here is an "error list" for the prep document of MoussorgskyOrchRavel_Tableaux_Gnome(57-109) (48 mes.):

- m.57 Perc.3: the expression « Bass Drum » is too high
- mm.57-61 Harp: the « 8va » are both too low - and is there a way to have them show « 8bassa » instead of « 8va »? (also m.66)
- m.65 indeed, the fermatas are misplaced… do you think an expression would have been better than an articulation?
- m.63 Bsn 1.2. Staccato on last quarter
- m.66 Cl. 2 & Bsn. 2 - no Dynamics?
- m.64 Vc. « Unis » collides with barline
- m.70 Fl. 1.2. & Ob. 1.2. « a2 » too high
- m.70 Cl.1.2. grace notes are not the same size?
- m.70 Bsn.2 « ff » too low
- m.66 all Horns: fermatas collide (is this because of transposition?)
- m.70 all Tpts: « Sourdines » misaligned with one another
- m.70 Vc. grace note wrong size
- m.73 Vc. last upper gliss. coincides with barline too much and is hard to see (is that something we can control?)
- m.72 Vla. « invisible gliss. »
- m.71 Vc. « invisible gliss. »
- m.71 Vc.b « invisible gliss. »
- m.70 Db. collision of expressions
- m.76 Ob. 1.2. & Cl. 1.2. grace notes wrong size
- m.76 Tbn.3 & Tuba « ff » a bit too low (esp. Tuba)
- m.76 Vl. I & Vl. II & Vla. & Vc. grace notes too big
- m.76 Vl. II « port » too high
- m.76 Db. half-rests separated
- m.75 Vc. & Vc.b « invisible gliss. »
- m.86 all flutes « mf » misplaced
- mm.82-83 B. Cl. hairpin-slur collision (because transposing?)
- m.83 Bsn.1.2. « 1. » misplaced
- m.84 Tbn.3 « mf » too low
- m.85 Perc.4 No font for xylophone? (also, misaligned)
- m.81 Vl.I « invisible » gliss.
- m.86 Vl.I & Vl.IIb « Pizz. » too high
- m.85 Vl.II expressions misplaced
- Vla m.80 « invisible gliss. »
- mm.83-86 Vc. expressions and hairpins a bit too low
- m.86 Db. hairpin slightly too low
- m.90 all Oboes « mf » too much to the right
- mm.87, 90-91 B.Cl slur collides
- m.87 Bsn.1.2. « 1. » misplaced
- mm.93, 95; Cl.1.2.: expressions misplaced
- mm.94, 96 Cl.2 expressions misplaced
- mm.94-95 B.Cl. hairpin collision (transposition?)
- m.95 Bsn.1.2. « mf » misplaced
- mm.96-97 Bsn.1.2. slur collides
- m.96 Bsn1.2. « tr » and « f » collide
- m.93 Vla. expressions misplaced
- m.100 Fl.1.2. two eight-note rests
- mm.102-103 combined Hn: « stop » expressions on wrong side
- m.103 Hn.2&4 Fermatas
- m.101 Brasses: some « Sourdine(s) » are in the wrong bar (I see they really need to be aligned at the very precisely good spot)
- m.103 Tbn.3 fermata collides
- m.103 Tpt.1.2. & Tbn.1.2. miss their fermata!
- mm.98-100 Harp: left hand rests are too high
- m.98 Vla. « f » too low
- m.109 fermatas…
- m.105 Fl.2 & Ob.2 slur collides
- m.108 Ob.2 & Cl.2 slur collides
- m.108 Hn.1.2. & Hn.3.4. « ff » misplaced
- m.104 Brasses: « ôtez… » expressons misaligned
- m.108 perc.3 « ff » misplaced
- Harp m.109 « ff » collides

8. Updated Note-Entry Protocol

With the new Note-Entry Protocol, most of the problems and steps outlined above will be taken care of automatically within the Note-Entry process. The new Note-Entry Protocol will remove the following steps from the XML Export Protocol:

- No need to delete hidden information/playback data
- No need to add percussion mappings
- No need to create the Separate/Combined versions of the score
- No need to correct any of the issues corrected by the "warnings" list below

We have also added the following "warnings" to the Note-Entry protocol:

- No, or minimal, adjustments of articulations
- No "double stems" instead of notes in two layers
- Fermatas must be written as *Expressions*, not articulations
- Do not use expressions that use two different fonts; instead, break them into two separate expressions
- Do not use expressions that use multiple lines of text; break lines into separate expressions
- Do not attach expressions to the end of an empty measure – always attach to the first note, or the barline, of the following measure
- Ensure that all expressions are aligned vertically (use alt + up/down to copy expressions to upper/lower staves)
- Do not displace vertical 2nds between two parts in the combined score that collide
- Do not use percussion graphic symbols (these can be included in the final pdf layout)

9. Conclusion and Future Developments

As of April 5, 2019, 47% of the total pieces in the OrchPlay Library (52% of the total measures) have been created as "prep" files, whereas 12% of the library has been fully completed for all four XML versions (20% of the total measures). Most of the time spent on the development of XMLs so far has been on automating the correction of recurring problems and creating a fast and efficient workflow which is beginning to take hold.

Since the addition of Dominique Lafortune to the XML production team, the process has sped up immensely and the quality of the files has likewise improved. We are confident that we can complete the production of XMLs for **the entire OrchPlay library by the end of May 2019**.

Moreover, with the implementation of the new Note-Entry Protocol, combined with the continually developing python scripts that help to automate the process, the new XML files created after May 2019 will take very little time, and be integrated into the process of Note-Entry seamlessly. We are eager to begin working with these files in OrchView and OrchPlay, and to see how the user community can employ these standardized files to advance research in the analysis, creation and teaching of orchestration.