KSN musical harmony annotation format

Revision 04/21/11

Table of contents

Introduction

There is currently no widely accepted standard for storing musical harmony information. Some people opted to use Harte's [ ] chord dictionary, which is quite large, but still not flexible enough for all purposes and it does not specify how to relate chords to the original musical score.

In this work, we have proposed a harmony annotation standard that covers virtually all aspects from chord definitions, keys, modulations, tempo, and score-to-label synchronization. The standard is meant to be used both by engineers and musicians and therefore easy to process automatically, at the same time easy to use by people not accustomed to computer technologies. The resulting annotation format is text-based and tries to immitate real musical notation where possible; in this it resembles tha popular ABC musical notation format [ ].

This work is the result of joint effort of Prof. Hitomi Kaneko of the Toho Gakuen School of Music and her students (Kojiro Ebiko, Yu Noda, Arata Yumi), and Prof. Shigeki Sagayama of the University of Tokyo and his students (Stanisław A. Raczynski, Daisuke Kawakami).

Notation format

Basic file structure

An KSN file contains harmonic annotation for a single piece of music. There are three main elements of the KS notation language: comments, directives and the harmony data.

Pitch naming convention

West-European pitch naming convention is adopted in the standard. Therefore, e.g. C-major scale contains pitches named C, D, E, F, G, A and B, as compared to C, D, E, F, G, A and H in the Central-European notation.

Comments

Comments begin with a percent sign % and end with a line break.

```
@K=C                % this is a comment
I | IV | V | I ||   % here's another comment
```

Directives

All directives start with the *at* sign @. Currently, the following directives are used:

	Function	Examples

@K	Specifies the key. A single letter (A, B, C, D, E, F, G) is used to determine the pitch class of the tonic (letter notation). Uppercase letters stand for major scales and lowercase letters for minor ones. Flats are specified with a minus sign − and the sharps with a plus sign +, placed in front of the letter.	`@K=+C   % C#-major`
		`@K=-d   % Db-minor`

@M	Specifies the meter (time signature). The first numeral defines the number of beats per measure, while the second one the note value that constitutes a single beat. They are separated by a slash sign /.	`@M=4/4`
		`@M=12/8`

@S	*Segno*.	
@F	*Fine*.	
@C	*Coda*.	
@DCAF	*Da capo al fine*.	
@DCAC	*Da capo al coda*.	
@DSAF	Da segno al fine.	
@DSAC	Da segno al coda.	

Table 1. List of directives.

There can be more than one directive specified in a single line, separated by white characters, they can also be mixed with the harmony data in the same line. A directive may appear anywhere in the file and applies only to the harmony data that stands after it. This will typically be used in situations when the meter changes throughout the piece or for polyrhythmic pieces.

Harmony data

The harmony data is basically a sequence of chords (tonal centers), bar lines, modulation indicators and repetition control marks. Like in ABC, bars boundaries are marked with pipe signs | and musical phrase boundaries can be indicated with double pipes ||.

```
@K=C               % in the key of C-major
I | IV | V | I ||  % phrase ends with an authentic cadence
V | V | I          % further three bars
```

Chords must be separated from other chords by at least a single space, but this does not apply to other language elements. All additional spaces are ignored.

Specifying chords

A chord definition consists of five parts, some of which are optional.

Element	Description	Examples	
Note value (*optional*)	An arabic numeral indicating note value relative to the current meter. The unit is not one beat, but depends on other note values in the current measure. If no note value is specified, a unit length is assumed.	3	Chord lasts for 3 length units.
Tonicization (*optional*)	Short-term (only for the duration of the current chord) modulation.	V:	Chord from the dominant key.
Root modificators (*optional*)	See *Modificators*.	+	Root is raised by a semitone.
Root specifier or member note list	Roman numeral indicating scale degree relative to the current scale or an absolute chord name (in letter notation) or a list of member notes enclosed in square brackets.	V [C E]	Dominant chord. Notes C and E.
Modificators (*optional*)	Series of marks that indicate raising, lowering, deletion or addition of tones to or from the chord. Tone in question is specified with an arabic numeral indicating number of scale steps, i.e. the generic interval from the root of the chord.	1!5+7	Root is missing, fifth is raised by a semitone and a seventh is added.
Inversion (*optional*)	A series of apostrophes ('); their number indicates the inversion degree.	'''	Third inversion.
Added notes (*optional*)	A series of added notes, each preceded by an ampersand.	&2 &C	Added second. Added C note.
Pedal note (*optional*)	A pedal point, i.e. single note in pedal position added to the chord, preceded by a slash.	/I	Added tonic in pedal position.

Table 2. Structure of a chord definition

There are three special symbols that can be used instead of the root specifier. Both do not allow for any modificators or inversion marks. They are listed in Table 3.

Symbol	Meaning
q	No chord. It can be used to pad measures in case of an *upbeat* (*anacrusis*, *Auftakt*) or when no chord is being played (drum solo).
z	*Rest* (no notes played).
_	Chord repetition mark (underscore). The previous chord declaration is repeated.

Table 3. Special chord markings.

Capital letters denote major chords, while lowercase letters minor chords. All other chord types are specified by modifying the note members with tone modificators.

Letters	Chord intervals (intervals)	Examples	
capital	root (+0), major third (+4) and perfect fifth (+7)	`@K=C`	
		`I`	C, E, G
lowercase	root (+0), major third (+3) and perfect fifth (+7)	`@K=+D`	
		`v`	A♯, C♯, E♯

Table 4. Two basic chord types.

Letters	Scale	Intervals between notes (in semitones)
capital	major (Ionian mode)	2, 2, 1, 2, 2, 2, 1
lowercase	natural minor (Aeolian mode)	2, 1, 2, 2, 1, 2, 2

Table 5. Two basic scales.

Note value

Note values are specified according to the current meter. The unit might be a beat, but it might also be a division of a beat. To calculate the note value in beats, the whole measure must be taken into account.

```
@M=4/4        % Note values in B (beats):
2I IV V |     %  2B (half note), 1B (quarter note), 1B
@M=12/8       %
2V7 !V9 I |   %  6B (dotted half note),
              %  3B (dotted quarter note), 3B
@M=2/4        %
5/4IV 3/4V |  %  5/4B (five sixteenth notes),
              %  3/4B (three sixteenth notes)
```

The number of beats for a chord can be calculated with the following formula:

$$B_i = N \cdot L_i / \sum_k^K L_k ,$$

where L_i is the note value specified for i-th symbol in the current measure, N is the number of beats per measure and K is the number of symbols in the current measure.

Extended chords

Extended chords are built by specifying the interval of the *highest* member note.

```
V   |  % major dominant chord
V7  |  % seventh dominant chord
V9  |  % ninth dominant chord
V11 |  % eleventh dominant chord
V13    % thirteenth dominant chord
```

All notes below the specified chord extension are assumed to be present. For example, "V13" contains all the root, the third, the fifth, the seventh, and so on up to the thirteenth.

Explicit member notes

Notes that belong to a chord may be specified explicitly using the square brackets operator [ ], which groups them into a single chord. They can be given either with Latin letters (absolute pitch class), or Roman or Arabic numerals (relative pitch class). They should be separated by spaces.

Just like with the key directive, flats are specified with a minus sign – and the sharps with a plus sign +, placed in front of the letter. Apostrophes can be used to specify notes from octaves higher that than of the root.

	Symbols	Meaning
letter notation	A, B, C, D, E, F, G	Absolute pitch class. Capital letters must be used.
arabic numerals	1, 2, 3, 4, 5, 6, 7	Pitch class relative to the chord root. The numeral indicates a generic interval above the root (1 = first, 2 = second, etc.).
roman numerals	I, II, III, IV, V, VI, VII	Pitch class relative to the tonic. The numeral indicates scale degree of the note (I = tonic, II = supertonic, etc.). Capital letter must be used.

Table 6 Three symbol sets used to implicitly specify member notes of a chord.

```
@K=D
[D +F A] |   % equivalent to I
[I III V] | % the same chord specified in relative terms
[I +F 5] |   % absolute and relative terms can be mixed
[V +C E'']   % E'' is two octaves higher than E
```

Redundand chord defitions

A single chord can be given in several equivalent forms interlaced with the equal sign =. This feature is mainly for the convenience of the human reader.

```
@K=C
V7=[G B D' F'] | IV''=[C F A]
```

Member note modificators

Each member note can be modified with one of the following modificators:

Modificator	Meaning	Examples	
+	Raising by 1 semitone	3+	raised third
++	Raising by a whole tone (2 semitones)	++F	F♯♯
–	Lowering by 1 semitone	1–	lowered root
––	Lowering by a whole tone (2 semitones)	––C	C♭♭
!	Deletion	5!	missing fifth

Table 7 Tone modificators.

The raising and lowering operators are generally placed before the operand with the exception of arabic numerals, which precede the operator.

```
@K=C
V3!7=[D F G] |
!V7=vii-=[VII II' IV']=[B D' F'] |
V!3!9=[V IV' VI']=[G F' A']
```

When modifying the root note, the modificator can be placed directly before the chord name. When modifying the fifth, the modificatr can be placed directly after the chord name.

```
!V |  % root is missing, shorthand for V1!
V+ |  % fifth is raised, shorthand for V5+
```

Modulations and tonicizations

A tonicization is indicated by specifying a temporary tonic center followed by a colon mark :. The relative notes in the chord that immediately follows the tonicization command will be regarded as relative to the temporary tonic center. The tonicization does not carry over the redundancy mark (=). The new tonic center can be extended to include many chords by using the curly brackets operator { }, effectively marking modulations.

```
% Bars 1-3 of Chopin's Nocturne No. 2
@K=-E @M=12/8
I !V9'''/I I I7''' | {ii: 2V7 !V9/i i} | V7 vi:V7' vi V:!V9'
% The second bar modulates to the supertonic and the third
% bar has two tonicizations: to the relative key and to the
% dominant
```

Added notes

Individual notes (added tones) can be added to a chord with ampersand operator & and they do not have to be enclosed in the square brackets. They can be specified either with:

1. latin letters (A, B, C, D, E, F, G),
2. arabic numerals (1, 2, 3, 4, 5, 6, 7),
3. or with roman numerals (I, II, III, IV, V, VI, VII), but then they must be enclosed in square brackets in order to avoid confusing individual notes with chords.

```
@K=C
I3!&2&4=[C D F G] | vi&[V]=vi&[G]
```

Pedal notes

If an added note is at the pedal position, the slash operator / can be used. This can be used to mark e.g. pedal point. Because only a single note can be added to the chord in this manner, the square brackets are not used.

```
@K=C
V/I=[C G B D']
V3!7/I=[C D F G]
```

A pedal point can be added to several chords at the same time using the grouping operator { } and the addition operator &.

```
@K=C @M=2/4
[8V]&{V7 I V vi | iii IV I V |}
```

Borrowed chords

Borrowed chords are market with parentheses around them. For example:

```
% The first 4 bars of Brahms' The Hungarian Dance No.5
% The piece is performed on two pianos and separate
% chords are specified
@K=+f @M=2/4
i (i'') | i (i'') | !V9'''/i (!V9''') | i (i'') |
```

Repetitions and jumps

The simplest way to specify repeats is to use the repetition operators: |:, :|, ||: and :||. These operators are used in place of the bar lines (|), so only entire measures can be repeated in this manner. To denote repetitions of measure fragments, a parenthesis versions of these operators are used: (: and :). Note that that the opening repetition mark may be skipped and in that case it is assumed to be placed at the beginning of the harmony stream. If a closing mark appears in the same place as an opening mark for another repetition, they can collapse to form a :||: mark. Alternative endings are marked using |[1 and :|[2 marks.

<table>
<tr><th>Repetition operator</th><th>Meaning</th></tr>
<tr><td>|: ... :|</td><td>Repetition of entire measures.</td></tr>
<tr><td>||: ... :||</td><td>The same as “|: ... :|”, but used in place of phrase boundary marks (“||”).</td></tr>
<tr><td>:||:</td><td>Equivalent of “ :|| ||: ”.</td></tr>
<tr><td>||: ... |[1 ... :|[2</td><td>Alternative repetition endings.</td></tr>
</table>

Table 8 Repetition operators.

```
 I | IV ||: V |[1 IV :|[2 I ||
% This is equivalent to:
 I | IV | V | IV | V | I ||
```

Additionally, repetition directives can be used to specify repeats and they take precedence over the repetition operators.

Directive	Corresponding musical term	Meaning
@S	segno (𝄋)	Marks the beginning of the repetition.
@C	*coda* (※)	Marks the end of the repetition.
@F	*fine*	Marks the ending of the piece after the second repetition.
@DCAF	*da capo* (D.C.) al fine	Repeat from the beginning until the end or until fine @F, if present.
@DCAC	da capo (D.C.) al coda	Repeat from the beginning until coda @C.
@DSAF	*dal segno* (D.S.) al fine	Repeat from the segno @S until the end or until fine @F, if present.
@DSAC	*dal segno* (D.S.) al coda	Repeat from the segno @S until the coda @C.

Table 9 Repetition directives.

Example annotation for the RWC database

As an example, let us observe the KSN labels for the file RWC-C24A:

```
%
% RWC-MDB-C-2001 No.24-1
%
@K=G @M=3/4
% bars 1-16
 I | I' | IV! | I!' | V3!7'' | I! | V! I!' I! | V3!7 | I' | I | IV! | I' |
2V3!7'' (!V') | 2I! I!' | ii!' 2V | I |
% bars 17-32
 I! | V' | 2vi! vi!7 | {V: V! | V | IV!' I! !V' | 2I V! } | V7 | I' | IV | !
I' ii! I! | V | V | IV!' I! !V' | I I!' V! | I ||
```

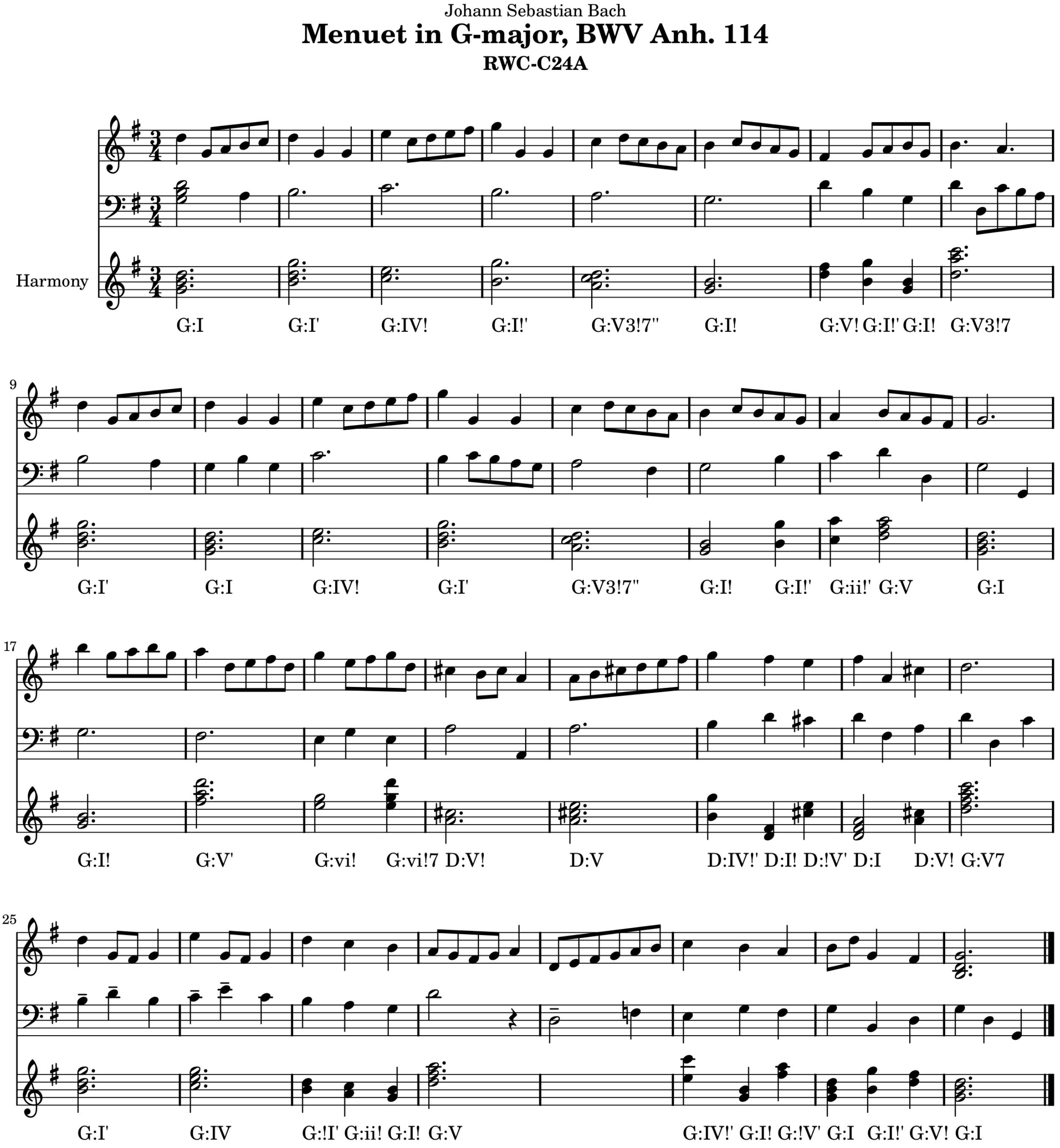
Johann Sebastian Bach
Menuet in G-major, BWV Anh. 114
RWC-C24A
Harmony
G:I G:I' G:IV! G:I!' G:V3!7'' G:I! G:V! G:I!' G:I! G:V3!7
9
G:I' G:I G:IV! G:I' G:V3!7'' G:I! G:I!' G:ii!' G:V G:I
17
G:I! G:V' G:vi! G:vi!7 D:V! D:V D:IV!' D:I! D:!V' D:I D:V! G:V7
25
G:I' G:IV G:!I' G:ii! G:I! G:V G:IV!' G:I! G:!V' G:I G:I!' G:V! G:I

Figure 1: An example score (RWC-C24A) annotated with harmony, generated from the corresponding KSN file above.

Table annotation format

The above annotation format is extremely flexible, but for many applications too complicated and requires a specialized parser. For that reason, the annotation can also stored in a simplified CSV[1]-based format. However, it must be emphasized, that tha table format cannot contain all the information of the KSN format (e.g. repetitions).

Basic data format

This format was devised based on the following assumptions:

- It should be a text format easily importable to Matlab, GNU R, MS Excel, OpenOffice Calc, etc.
- It should contain as much information compared to KSN as possible.
- It should only use numeric values for ease of processing, i.e., all elements should either be real numbers or missing values (NA/NULL).

A basic set of fields is described in the table below:

	Field name	Description
Time	Measures	Length of the chord in bars.
	Beats	Length of the chord in beats.
	Ticks	Length of the chord in ticks. 48 ticks = a whole note.
Key	Signature	Number of flats (negative values) or sharps (positive values) in the key signature. -7,...,0,1,...7
	Mode	Musical mode. 0 for major (Ionian), 1 for minor (Aeolian) mode.
Chord	Degree	Scale degree. A value between 1 (tonic) and 7 (leading note) or NA for non-chord.
	Type	Chord type. 0 for major and 1 for minor.
	Inversion	Chord inversion. 0, 1, ...
	Root	Root modification. 0 for no change, non-zero values for lowering or raising by the given number of semitones, and NA for a missing root.
	Second	Second note modification. See root modifications.
	Third	Third note modification. See root modifications.
	Fourth	Fourth note modification. See root modifications.
	Fifth	Fifth note modification. See root modifications.
	Sixth	Sixth note modification. See root modifications.
	Seventh	Seventh note modification. See root modifications.

---

[1] CSV — Comma Separated Value (text-based format supported by most data-processing environments).

Field name	Description
Ninth	Ninth note modification. See root modifications.
Eleventh	Eleventh note modification. See root modifications.
Thirteenth	Thirteenth note modification. See root modifications.
Fifteenth	Fifteenth note modification. See root modifications.
Added	Added note. Absolute pitch class, 0 for C, 1 for C#, etc. or NA if not present.
Pedal	Pedal note. Absolute pitch class, 0 for C, 1 for C#, etc. or NA if not present.

Ticks are 1/96th of a whole note. This number has been chosen to avoid fractional lengths. Of course it is still possible (though highly improbable) to have fractions, and therefore a smaller tick value may be considered.

Three length descriptors are used in order to be able to restore the time signature $\frac{M}{B}$:

$$M = \frac{Beats}{Measures}, \quad B = 96\frac{Beats}{Ticks}.$$

It might seem redundant to include the length descriptors for each line, but the time signature can change throughout the piece of music, and it can easily be tracked this way.

The key (current tonal center) is represented by a signature and a mode (in contrast to tonic + mode representation) to retain the enharmonically equivalent keys (like B♮-major and C♭-major).

Extended data format

The data format can be supplemented with redundant fields for easier processing.

	Field name	Description
Time	Measure sum	Cumulative sum of lengths in bars. This holds information about the beginning of the chord.
	Beat sum	Cumulative sum of lengths in beats.
	Tick sum	Cumulative sum of lengths in ticks.
	Beats per measure	Number of beats in one measure (upper part of the time signature).
	Ticks per beat	Note value of 1 beat, in ticks (lower part of the time signature).
Key	Tonic	Tonal center of the key. Absolute pitch class: 0 for C, 1 for C#, etc.
Chord	Absolute root	Chord root as an absolute pitch class: 0 for C, 1 for C#, etc.

Parser implementation in GNU R

A parser of KSN files able to output the above format has been implemented in R.

Function name	Description
`readKSFile`	Loads a KSN file as a table with 5 columns: length in ticks, key (in KSN notation), chord (in KSN notation), beats per measure, ticks per beat.
`interpretKSKey`	Interprets KSN key and returns either a key signature + mode, or tonic + mode pair.
`interpretKSChord`	Interprets KSN chord and returns a tuple containing: scale degree (1..7, NA for non-chord), chord type (0 for major, 1 for minor), chord inversion (0,1,...), chord member note modifiers (NA for missing note, 0 for present note, +1, -1, +2, -2 for raised or lowered note) for the root, 2nd, 3rd, 4th, 5th, 6th, 7th, 9th, 11th, 13th and 15th, and added and pedal notes (NA for no added note or a number between 0 and 11 corresponding to an absolute pitch class).
`readKSFileAsTable`	Reads a KSN file and converts it into a table format described in the first section of this document.
`saveKSAsTable`	Saves a harmony table obtained with `readKSFileAsTable()` to a text file.
`Ksn2midi`	Reads a KSN file and saves it as a MIDI file. The MIDI file contains a single track with chords played with accompanying chord labels.

Database structure

Folder name	Contents
Code	Parser implementation in GNU R and auxiliary scripts.
Documentation	This documentation.
RWC-Classical	Files corresponding to the *classical* portion of the RWC database.
RWC-Genre	Files corresponding to the *genre* portion of the RWC database.

Table 10 Structure of the database package.

Folder name	Contents
Harmony	Harmony annotation data.
KSN	In KSN format.
CSV	Converted to the table format.
MIDI	Converted to MIDI files.

Table 11 Structure of each of the RWC- folders.*