This material is based upon work supported by the National Science Foundation under Grant No. 9909068. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

A Music Representation Requirement Specification for Academia

Donald Byrd, Senior Scholar

Eric Isaacson, Associate Professor of Music Theory

School of Music, Indiana University

(from *Computer Music Journal* 27(4), 2003; last revised early May 2005)

Introduction

In General

The published literature on music representation is substantial. In addition to works on Common (Western) Music Notation (CMN) addressed to musicians (Read 1969; Stone 1980), there are now papers and books intended for programmers interested in music and for researchers (Byrd 1984, 1994; Dannenberg 1993; Wiggins 1993; Selfridge-Field 1997; and Hewlett and Selfridge-Field 2001). Yet we know of nothing like a detailed and comprehensive description of the requirements for a music representation in any situation. As a result, developers both of music-editing programs and of music representations have always been "on their own."

The Indiana University School of Music is one of the world's largest music schools. *Variations2* is a large-scale digital music library project under development at Indiana University. Version 1 of the software for *Variations2* is now complete; for that version, the digital library contains music in only audio and score-image forms. This document lays out requirements for a symbolic music representation to be used beginning with a future version of *Variations2*. Symbolically represented music in *Variations2* will be of interest to a wide range of people, for a wide range of applications. This will include:

(1) music faculty, especially in music theory and music history, who are creating assignments and teaching classes (for showing and playing musical examples and analyses of those examples);

(2) students enrolled in classes with these faculty (for doing assignments);

(3) both faculty and student music researchers doing content-based analytical or historical research;

(4) a minority of other music library patrons who, for whatever reason, are not content with scanned scores.

This specification was originally intended for use in developing *Variations2*. It reflects the fact that Indiana University School of Music is heavily oriented towards "classical" music (western art music), though it also has a strong program in jazz and offers courses in popular music. We believe, however, that our requirements are similar to those of almost any academic music department with a similar emphasis on "classical" music. Specifically, we believe that most music departments that emphasize classical music will have similar requirements regardless of how they approach teaching music theory and analysis and—at least within the limits of music for performance by instrumentalists and singers—regardless of what styles of composition they emphasize. Beyond that, these requirements directly reflect what information is important in notating music, and they should therefore be of considerable interest to designers of music-editing programs.

Wiggins et al. discuss three sorts of tasks a symbolic music representation might be used for (Wiggins et al. 1993): recording, where "the user wants a record of some musical object, to be retrieved at a later date"; analysis, where the user "wants to retrieve not the 'raw' musical object, but some analyzed version"; and generation/composition. Of these, we are concerned most with the first, less with the second, and least with the third. Declarative representations—by far the more familiar type to most people—rather than procedural ones are much more appropriate for the first type of task and usually

for the second, and accordingly we consider only declarative representations here. Nonetheless, we believe a representation that satisfies our requirements will also fulfill the needs of a great many composers and arrangers.

The primary use of the music representation at Indiana University will be to encode existing scores in CMN. This includes western art musics from roughly 1600 to the present, including standard twentieth-century works. We also want to be able to encode modern transcriptions of medieval and Renaissance works, plus jazz and popular music. Music in which the arrangement of graphical elements themselves is considered a part of the composition (*Augenmusik*) will not be encoded, nor will twentieth-century scores with substantial graphical representations (e.g., pseudo-scores representing electronic compositions, and graphical scores such as those by LaMonte Young or John Cage). On the other hand, we may like to encode tablature in music for lute, popular guitar music, and perhaps harp, but this is not essential, and specific requirements for it are yet to be written.

Despite the exclusion of generation/composition tasks, the expected users of the representation have a very broad range of interests both in terms of musical repertoire and in terms of what they wish to do with that repertoire. Clearly, the representation must be flexible enough to handle the wide range of variation found in the repertoire as well as in the likely uses. To give one example, it must be straightforward to represent music in which the durations of measures do not agree with the time signatures, and in which voices are synchronized in complex ways. It is not always easy, even for an expert, to tell what the duration of a measure is or exactly which notes are synchronized: there may be unmarked tuplets, or voices entering or leaving. Also, is not unusual to find music in which the cumulative note durations in a measure do not agree with the time signature, because of things like cadenzas and cadenza-like passages and mistakes by the composer or publisher. Finally, the representation must support such tasks as students doing melodic dictation and instructors creating deliberately incomplete or

incorrect notation examples for use in class or on student assignments. (This raises issues for playback timing, but they should not be too difficult to address.)

Another area in which much flexibility is desirable is in the relationship between pitch notation and sounding pitches. In older editions, French horn parts in bass clef are usually written an octave lower than their transposition would dictate, while timpani parts are written without key signatures or accidentals. Peculiarities of pitch notation also exist in older editions in bass clarinet and cello parts and others (Byrd 2004). Some of our users will be interested in the notation only, but many are likely to be concerned with the sounding pitch level as well; see Representing Pitch below for more discussion.

Note that this is a *requirements* specification, not a *design* specification: we wish to choose an existing representation rather than design a new one from scratch. This preference is reflected in many details of this specification. For example, it would be nice simply to decree that the representation must support scores of, say, 150 staves and thereby cover any remotely-reasonable eventuality, and it would surely not be hard for most representation designers to satisfy such a requirement. But supporting so many staves is of no importance whatever for existing music, and we cannot afford to significantly downgrade a representation that supports "only" 90 (the number listed as "desirable" in Item 1.5). As of this writing, the requirements outlined in this document are being used to evaluate candidate representations for *Variations*2. Developers have been invited to demonstrate how their representations meet the requirements, and to describe what changes they are willing to make to bring the representation into agreement with the requirements. (Castan 2003 provides the most complete list of existing music representations we know of.)

Some of the numerical requirements in a specification like this are inevitably somewhat arbitrary, particularly in the case of larger numbers. There is little doubt that the number 2 in the requirement for augmentation dots on notes (Item 4.6) is exactly

what it should be; the same cannot be said of the number 500 in the requirement for starting measure number (Item 7.11).

What is Covered

This specification was written with an eye toward supporting reasonably complete, independent descriptions of notation and MIDI performance. This is largely because it is so useful to be able to represent MIDI files with no notation information present and notation files with no performance information present, though of course either can (and no doubt often will) be crudely inferred from the other. (Huron 1997 describes this idea in general as "selective feature encoding.") It is also important for our project that we be able to represent a score and a musical (i.e., audio) interpretation of it with synchronization at the measure level, or a rough equivalent for music without measures. Synchronization at the note level would be even better. A related need is to be able to navigate (presumably via a GUI of some kind) from the image form of a score to the symbolic representation and back, though we have no specific requirements here.

Domains of musical information. The groundbreaking Mockingbird music editor (Maxwell 1981; Maxwell and Ornstein 1984) pioneered the approach of storing independently information about the *logical, performance*, and *graphic* aspects of music. The logical domain describes a piece of music as the composer might think of it; the performance (also called *gestural*) domain describes it as sound waves or manipulations of a musical instrument; the graphic (also called *visual)* domain, as a collection of marks on paper. Logical information about a note might include that it is a dotted-quarter note; performance information, that it lasts for 684 ticks; and graphic information, that it has an open-diamond-shaped notehead and a stem extending upward for 360 twips (20ths of a point). Many symbols of music notation—beams, ties, octave signs, etc.—exist only in the graphic domain. NIFF (Grande 1997), as well as Nightingale (AMNS 2002) and other programs, adopted Mockingbird's approach; SMDL (Sloan 1997) added a fourth

"domain," for *analytic* information: information about the work, which might include bibliographic information, as well as interpretive information that ranges from phrase markers and roman numeral analysis of underlying harmony to a Schenkerian graph. We use and strongly advocate SMDL's version of this independent-domain model.

Why have a logical domain? It can be argued that the very concept of a logical domain is flawed, that one should simply encode the marks on the page as the best indication of the composer's intent. The basic issue here is, when should the marks be interpreted, at the time of encoding or the time the data is used? This is a serious issue, and we cannot do justice to it here. But, at least in our setting, we feel it is better done at the time of encoding. By far the most common example of a situation where the graphics alone fail to capture an important aspect of the meaning of the music is the invisible tuplet. Where long series of similar tuplets—nearly always triplets—occur in succession, music from the late Baroque on omits the tuplet marking after the first instance or two; in fact, cases where the marking does not appear on any of the instances are not too unusual. Of course it may be possible for a computer program as well as a human being to infer the presence of the tuplets. Horizontal alignment is usually a strong clue, but by no means always; the same can be said of beaming, and even of the total duration of notes in each voice in a measure as compared to the time signature. But should every program that wants to do anything nontrivial with the music be required to include logic to consider alignment, beaming, and durations within voices—not to mention deciding voicing in the many ambiguous cases? Clearly not.

Actually, a similar argument could be made against the performance domain. These arguments lose most of their force if it is acknowledged that, for almost all classical music, the graphic domain is the most objective representation of what the composer "wrote", and therefore the fundamental one in any encoding of music notation that claims to be authoritative.

Our focus and its implications for domains. We should emphasize that we are not interested in producing publishable scores, just serviceable renderings of the notation. So the graphic domain is perhaps least important for us of the four. But, as one of the present authors made clear in his dissertation (Byrd 1984), rendering complex music in a merely serviceable way is far more difficult than one might think. With current music-notation technology, even relatively simple music often needs tweaking we can ill afford to lose when there are two or more voices on a staff. (Powell 2002 describes the limitations of well-known programs in some detail.) Thus, it will be very helpful if our representation is capable of storing actual positions for symbols (preferably relative to their contexts, not fixed with respect to the page) and their sizes and shapes. Otherwise, we might start with a version of a densely-contrapuntal piece—perhaps scanned in from a published edition—in which the notehead, rest, beam, and dynamic positions, slur and tie shapes, etc., have been carefully tweaked for readability, but be forced to throw the tweaking away. For academic purposes, another serious argument for storing graphic information is in the display of Schenkerian notation, where standard position and shape rules do not apply.

Note that several of our requirements—symbols in parentheses, accidentals small or above notes, etc.—are stated in terms of graphics, but in many cases are clearly expressions of semantics, usually editorial additions. For these items, we also require it be possible to express the semantics. We feel it is best to represent the semantics explicitly when they are clear, but to represent only the graphics when they are not: we want the encoder to be able to choose either.

Finally, speaking of purely-graphic information, it might be asked, why is some such information covered while much (page numbers, page sizes and orientations, edition or plate numbers, etc.) is not? Again, we are interested in serviceable notation, not publication quality, so we try to include everything that might affect the readability of the music; information beyond that—while it might be essential to publishers or

historical musicologists—is beyond our scope. Of course, there is nothing to keep a representation from including the additional information. We include enough bibliographic information to identify the work clearly, but omit dates of composition and publication, provenance, etc.

Levels of Importance

We distinguish three levels of importance of features herein: *required, very desirable,* and *desirable*. Three is not a magic number of levels, but it seems most appropriate for our purposes. One advantage of distinguishing several levels is that it lets us soften any difficulties our subjective judgments of boundaries might cause: if starting measure numbers, say, of 600 or 700 turn out to be more important than we thought, at least these numbers will be available in a representation that supports the "very desirable" level of that feature. We have tried to accommodate through these levels the "extremes of conventional music notation," a handy compilation of which can be found in Byrd (2003).

What Does It Mean To Support Something?

What we mean by "support" for a feature may not be obvious: it often involves representing one or more relationships. For example, Item 3.6 says that noteheads, among other symbols, can be specified as in parentheses. This does not mean simply that it must be possible to say there are parentheses at a certain graphical position, a position which will result in those parentheses surrounding a particular note. Instead, the fact that the parentheses are around that note must be represented. We say nothing about what a program should do with the parentheses, e.g., if the note is transposed, moved to another staff, or deleted. But it is important for a program to know (without having to infer it from graphic or other information, a process that can be slow and

unreliable) that the parentheses are connected to that note so it can take what it considers the appropriate action.

Additional Considerations

For the representation we select, the existence of some sort of *schema* is necessary. A schema "is a formal definition of what comprises a valid document" (Harold and Means 2002). Schemas are important because they allow automatic validation of data and promote forward compatibility, both of which are important for our purposes. They also discourage undocumented extensions. Undocumented extensions, by a representation's creator or others, can wreak havoc on interoperability. We prefer representations whose developers avoid and discourage them. On the other hand, we have no problem with "official" extensions, presumably made available under the same terms as the format itself. Examples of acceptable schemas include XML Document Type Definitions (DTDs) and schemas (Harold and Means 2002) and Backus-Naur Form (BNF) descriptions.

In the vast majority of cases, CMN (unlike tablature) represents music in a descriptive rather than a prescriptive way (Seeger 1958). Where an aspect of the music can easily be represented either way, we always prefer the descriptive way because it avoids confounding the essence of the music with details of performance that are almost always irrelevant for our purposes. Pitch is the aspect this applies most clearly to: it can be represented in either sounding (descriptive) or written (prescriptive) form and, indeed, we prefer sounding pitch for transposing instruments, artificial harmonics, and even scordatura. However, pitch representation is a complex issue, and it is best by far to represent it in both forms. This point bears some discussion.

Representing Pitch

There are many instances in which written and sounding pitch are different, that is, instances in which "transposition" is used. The term is ordinarily taken in the sense of

"transposing instrument", i.e., a change in pitch that is consistent over a relatively long period of time, and that generally produces a change of key and of note name (e.g., clarinet in B-flat changes written C's to sounding B-flats) (Arnold 1983). But we find it useful to use the term transposition to refer to *all* differences between written and sounding pitch. For instance, transposition by an octave is found in standard notation for instruments like piccolo and double bass. However, there are much more subtle instances of differences between written and sounding pitch: implied accidentals in 18th- and early-19th-century timpani parts; clef-dependent octave shifts in older editions for horn, cello, etc.; the use of organ registration to change the pitch of a note; and—the extreme case—scordatura. With scordatura, the difference between written and sounding pitch can vary even from note to note of a single chord. Byrd (2004) lists the more complex cases, including some where the difference between written and sounding pitch is surprisingly difficult to discern.

The relationship between written pitch, sounding pitch, and transposition can be defined succinctly. By "written pitch" we mean the notated pitch, taking into account chromatic alterations (from the key signature and accidentals) and the effect of octave signs. The transposition (t) is simply the interval from the written pitch (w) to the sounding pitch (s). The relationship can be expressed in three ways:

[1] $s = w + t$

[2] $t = s - w$

[3] $w = s - t$

Note that so long as a representation includes two of the three pieces of information, the third can easily be computed.

Definitions

For clarity, we define a small number of terms here. All other musical terms in this document have their standard meanings.

Cable: in MIDI systems, a number used to allow addressing more than the 16 channels MIDI defines; it may or may not correspond to a physical cable. Each cable supports 16 channels independent of the others.

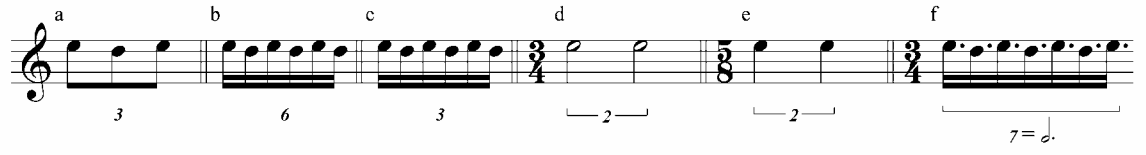
Chord: two or more simultaneous notes in a single voice (and therefore on one stem, unless no stem is present); the notes need not all be on the same staff. With this definition, a pianist might play two or more chords simultaneously. Also, the violinists in a string quartet might each play a chord, but they could not play notes of a single chord. (Note that this use of the word "chord" is that of nearly all music-processing programs, but is much more specific than its standard musical meaning of a harmonic entity that spans the full musical texture.)

Duration unit: in a tuplet, the duration to which the tuplet's numerator or denominator refers. In the vast majority of cases, the numerator and denominator duration units are the same. These values are derived from our colloquial descriptions of tuplets, for example, "Three eighth notes (the numerator and its duration unit) in the time of two eighth notes (the denominator and its duration unit)." Incidentally, the term "duration unit" is our own. Unfortunately, there is no standard term for any aspect of tuplets, including the word "tuplet" itself: terms such as the common "irrational group" make little sense. Furthermore, most discussions of tuplets by musicians are turgid and confused. A case in point is Read (1978). This is a generally first-rate book, and the extensive discussion of tuplets is filled with interesting comments and examples; but it is seriously lacking in clarity in this area.

In terms of the confusion they cause—particularly with respect to terminology—tuplets are in a class by themselves, so it is worth considering several examples (see Example 1). In order of increasing complexity: (a) A triplet containing 3 8th notes, labeled "3", and with a total duration of a quarter note has a numerator of 3, an

(implied) denominator of 2, and numerator and denominator duration units of an 8th note. (b) A tuplet containing 6 16th notes and labeled "6" has a numerator of 6, a normal denominator of 4 (but this is affected by the total duration), and duration units of a 16th note; (c) if the same tuplet is labeled "3," it has a numerator of 3, a denominator of 2, and duration units of an 8th note. (d) A tuplet of two half notes filling a measure of 3/4 and labeled "2" (as in Mahler's *Das Lied von der Erde*) has a numerator of 2 with duration unit of a half note, and a denominator of 3 with duration unit of a quarter. It could also be described by a denominator of 1 with duration unit of a dotted half. (e) A tuplet containing 2 quarter notes filling a bar of 5/8 and marked "2" (as in the third movement of the Barber Piano Concerto) has a numerator of 2 with duration unit of a quarter note, and a denominator of 5 with duration unit of an 8th note. It could also be described by a denominator of 1 with duration unit of 5 8th notes. (f) A tuplet containing 7 dotted 16ths filling a bar of 3/4 and labeled "7 dotted 16ths = dotted half" (in the Carter Concerto for Orchestra) has a numerator of 7 with duration unit of a dotted 16th, and a denominator of 1 with duration unit of a dotted half note.

Example 1



Part: We specify that a part may be *logical* or *analytic*. A logical part represents the music a single performer or closely related group of performers plays (or sings). Usually used for ensemble music, e.g., the 2nd violin part or bells-and-cymbals part of an orchestra piece, but the term applies even for solo music: a piano or unaccompanied harp piece has one part, which is identical to the score. If a group of performers does not play the same notes for the entire piece—e.g., strings in a piece with divisi sections—the term still applies as long as they play, or would normally play, from the same printed

music (perhaps with systems alternating between single and multiple staves). (In reality, this is not always well-defined: in an orchestral piece with two trumpets, the trumpets are likely to share a staff in the score, but the players might or might not play from separate first and second trumpet parts.) An analytic part contains notation that is not part of the music *per se*, but rather contains analytical information only. One example would be a Schenkerian graph. Either type of part can contain symbols of any kind; however, all symbols in an analytic part are considered analytic. Thus, if the score is played back, notes in an analytic part should not ordinarily be played.

Pseudobarline: A symbol that looks like a barline but does not function as a measure delimiter. Such symbols include mid-measure double barlines, repeat bars, and dotted barlines. The distinction is important for, at a minimum, understanding rhythm and numbering measures.

Sounding pitch has its usual meaning, except that we take it to include spelling, so MIDI note number alone does not capture its full meaning here. C#4 and D-flat-4 are different sounding pitches, even though both are the same key on a piano and the same MIDI note number (61).

Transposition is simply the difference between sounding pitch and written pitch. Since both involve musical notes, not just MIDI note numbers, a transposition is a musical interval: an instrument in D-flat transposes up a minor second; one in C#, if it existed, would transposes up an augmented unison. Note that, as described before, this is much broader than the usual definition. Under our definition, scordatura is the extreme case: to handle it, it is possible that every note of a single chord will have a different transposition.

Voice: a single "line" of music within a single part; it may contain chords, whose notes will almost always share the same stem. Simultaneous series of stem-up and stem-down notes or chords on a staff are considered separate voices. Some durations have no stems, however. This is a simple example of why the designation of voices will

sometimes require interpretive decisions by the encoder, but more difficult cases can certainly be found, especially in keyboard music, and most especially since Beethoven.

Written pitch is not just a note's position on a staff considering the clef, but the pitch as it is thought of by a performer, taking into account the clef, chromatic alterations from key signature and accidentals, and any octave sign.

Acknowledgements

Thanks to Jim Halliday for general discussion, and for suggesting we include definitions. Gerd Castan pointed out the importance of schemas, among other things. Thanks also to Tim Crawford, Marlin Eller, Michael Good, John Howard, Douglas McKenna, Donncha Ó Maidín, and Perry Roland for helpful comments of various kinds. Finally, feedback from the anonymous referees was exceptionally valuable.

This material is based on work supported by the National Science Foundation under Grant No. 9909068.

References

Arnold, Denis, ed. (1983). *The New Oxford Companion to Music.* Oxford: Oxford University Press.

Byrd, Donald. 1984. *Music Notation by Computer.* Ph.D. diss., Computer Science Department, Indiana University.

Byrd, Donald. 1994. "Music-Notation Software and Intelligence." *Computer Music Journal* 18(1): 17–20.

Byrd, Donald. 2004. "Extremes of Conventional Music Notation." http://mypage.iu.edu/~donbyrd/CMNExtremes.htm (18 December 2004).

Byrd, Donald. 2004. "Written Vs. Sounding Pitch. "

http://variations2.indiana.edu/system_design.html (August 2004).

Castan, Gerd. 2003. "Musical notation codes." http://www.music-notation.info/en/compmus/notationformats.html (31 December 2004).

Dannenberg, Roger. 1993. "Music Representation Issues, Techniques, and Systems." *Computer Music Journal* 17(3): 20–30.

Grande, Cindy. 1997. "The Notation Interchange File Format: A Windows-Compliant Approach." In Selfridge-Field (1997).

Harold, Elliotte Rusty, & Means, W. Scott. 2002. *XML in a Nutshell.* 2nd ed. Sebastopol, Calif.: O'Reilly.

Hewlett, Walter. 1997. "MuseData: Multipurpose Representation." In Selfridge-Field (1997), pp. 402-450.

Hewlett, Walter, & Selfridge-Field, Eleanor, eds. 2001. *The Virtual Score: Representation, Retrieval, Restoration (Computing in Musicology* 12). Cambridge, Mass.: MIT Press.

Huron, David. 1997. "Humdrum and Kern: Selective Feature Encoding." In Selfridge-Field (1997), pp. 375-401.

Isaacson, Eric, et al. 2002. Working documents for the Multimedia Music Theory Teaching (MMTT) project. http://theory.music.indiana.edu/mmtt/int (30 December 2003).

Maxwell, John Turner, III. 1981. Mockingbird: An Interactive Composer's Aid. M.S. thesis, Department of Electrical Engineering and Computer Science, MIT.

Maxwell, John Turner, III & Ornstein, Severo M. 1984. "Mockingbird: A Composer's Amanuensis." *Byte* 9(1): 384-401.

Powell, Steven. 2002. *Music Engraving Today: The Art and Practice of Digital Notesetting.* New York: Brichtmark.

Read, Gardner. 1969. *Music Notation*. 2nd ed. Boston: Crescendo.

Read, Gardner. 1978. *Modern Rhythmic Notation*. Bloomington: Indiana University Press.

Roland, Perry. 1997. "Proposed Musical Characters in Unicode." In Selfridge-Field (1997), pp. 553-564.

Schaffrath, Helmut. 1997. "The Essen Associative Code: A Code for Folksong Analysis." In Selfridge-Field (1997), pp. 343-361.

Seeger, Charles. 1958. "Prescriptive and Descriptive Music Writing". *Musical Quarterly* 44,2, pp. 184-195.

Selfridge-Field, Eleanor, ed. 1997. *Beyond MIDI: The Handbook of Musical Codes*. Cambridge, Mass.: MIT Press.

Sloan, Donald. 1997. "HyTime and Standard Music Description Language: A Document-Description Approach." In Selfridge-Field (1997), pp. 469–490.

Stone, Kurt. 1980. *Music Notation in the Twentieth Century: A Practical Guidebook.* New York: W. W. Norton.

Unicode Consortium. 2005. "Musical Symbols: The Unicode Standard, Version 4.1." http://www.unicode.org/charts/PDF/U1D100.pdf (1 April 2005).

Wiggins, Geraint, Miranda, Eduardo, Smaill, Alan, & Harris, Mitch. 1993. "A Framework for the Evaluation of Music Representation Systems." *Computer Music Journal* 17(3):31–42.

Appendix: Table of Requirements

We abbreviate the four SMDL domains thus: L=Logical, P=Performance, G=Graphic, A=Analytic. It is not always easy to say which domain or domains a feature belongs in, and some of the labels below are debatable. Also, Logical-domain items often have implications for the Performance and Graphic domains; these are not shown in the table. See Byrd 2003 for details of the many notational extremes (largest numbers of staves and voices, most augmentation dots, highest and lowest notes, etc.) we refer to. Each feature is assigned a priority as follows: Req = Required, VDes = Very Desirable, Des = Desirable.

No.	L	P	G	A	Pri.	Description
0. Global Information						
0.1				x	VDes	Supports notes about the music (e.g., explanation of non-standard notation or editorial details) preceding or, rarely, following the score.
0.2				x	VDes	Offers a way to describe the completeness of encoding, e.g., are editorial markings, beams, bowings, or even dynamics excluded; is MIDI info included or not.
0.3				x	Req	Provides the following metadata: • Composer(s) • Title of work
					VDes	Additionally provides the following metadata: • Source(s) • Other Creator(s) • Encoder(s), with indication of which domain(s) each is responsible for • Catalog (opus or other) number(s) • Uniform Title
1. Voices, Staves, and Parts						
1.1	x			x	Req	Voices belong to parts (as described in Definitions above).
1.2			x		Req	A voice can wander freely among the staves of its part.
1.3					Req	A voice can appear or disappear at any point, not just at measure boundaries (for example). Furthermore, all voices can disappear simultaneously, resulting in "incomplete notation" (presumably shown with blank space).
1.4				x	VDes	A part may be designated as belonging to the analytic (as opposed to the logical) domain.

No.	L	P	G	A	Pri.	Description
1.5			x		Req	A part can share a staff with related parts (e.g., 1st and 2nd bassoons, or 2nd and 4th horns), temporarily or for the entire score, if neither part is in the analytic domain.
1.6	x			x	Req	Support for 60 voices, 40 parts.
					Des	Support for 90 voices, 90 parts. (Stravinsky's *Le Sacre du Printemps*, with its large orchestra, uses a maximum of about 38 notated voices and 35 staves; the largest numbers we know of in published music are in a work of Ligeti that uses about 79 voices and 80 parts, on 77 staves.)
1.7			x		Req	Support for 40 staves.
					Des	Support for 90 staves.
1.8			x		Req	Supports up to 4 staves per part at any point.
					Des	Up to 8.
1.9			x		Req	Supports up to 3 voices per staff.
					VDes	Up to 4 (e.g., for the keyboard version of the Six-Part *Ricercare* from Bach's Musical Offering, or, in standard editions but with less justification, Bach's solo-violin Chaconne).
1.10			x		Req	Full displayed names of parts—i.e., for instruments or (singing) voices—can have up to 30 characters.
					VDes	Up to 50 characters. (The longest part name we have seen, in a work of Ives, is 47 characters.)
1.11			x		VDes	Short (abbreviated) displayed names independent of the full names can have up to 15 characters.
					Des	Up to 20 characters.
1.12			x		VDes	The name of a part can change at any point (e.g., "flute 3" to "piccolo").
1.13	x				VDes	Parts have internal names, independent of those displayed, of up to 20 characters.
1.14	x				Des	Internal names of parts adhere to some standard but user-definable vocabulary. (Internal names of parts are closely related to General MIDI program numbers as well as to Device Names and, especially, Program Names in standard MIDI files.)
1.15	x				Des	Supports optional passages (normally shown with symbols smaller than usual for the staff size).
1.16			x		Des	Supports cues (normally shown with symbols smaller than usual for the staff size). (Notice that cues involve not only notes, but also can include chords, beams, rests, clef changes, etc.—anything that can appear in a voice or affect a voice.)
1.17		x			VDes	Supports transposition changes. (This is not Required because its most important implications are covered in Section 4.)
1.18			x		Des	For scordatura, the "accord" (tuning to be used) is encoded explicitly. (Scordatura for string instruments is a particularly difficult form of transposed notation because it effectively involves different transpositions mixed within a single part, and even within a single chord. Its performance implications are covered in Section 4.)
2. MIDI Channels, Cables, and Patches						
2.1		x			Req	The patch any MIDI channel is set to can be specified (using standard patch addressing schemes, including General MIDI); if notes can be assigned to cables (see 2.3), it must be possible to specify the patch to which any

No.	L	P	G	A	Pri.	Description
						combination of channel and cable is set.
2.2		x			VDes	Patches can be changed at any point during playback. (These changes might well be associated with changes of part name, and of course the "arbitrary MIDI data streams" discussed under Miscellaneous Performance Elements would handle them.)
2.3		x			Req	Supports 16 MIDI channels.
					VDes	Effectively supports 64 MIDI channels (presumably via multiple cables).
					Des	Effectively supports 128 MIDI channels (presumably via multiple cables).
3. Musical Symbols in General Domain labels do not apply and are deliberately omitted for Items 3.1 through 3.4.						
3.1					VDes	The representation of musical symbols is reasonably intuitive.
3.2					Des	The representation is not too verbose. (While necessarily subjective, we would suggest, with "natural" formatting, no more than 4 lines and 100 characters per average note, and less if possible.)
3.3					VDes	The representation is extensible in a backwards-compatible way.
3.4					VDes	There is an "unknown" value for every field of every symbol. (This is necessary because of our desire to allow independent descriptions of multiple domains; information may sometimes exist only for one, but not the other, domains. Notice, however, that fields are not always logically independent: for example, it does not make much sense for a note of unknown duration to have two augmentation dots.)
3.5			x		Req	Horizontal position offset (relative to other notes/rests with the same logical onset time) can be specified for notes and rests. (Two voices on a staff is quite common, but few existing notation programs do a good job of automatically positioning notes—much less associated beams, ties, and other markings—in such cases: see the chapter on divisi notation in [Powell 2002]. Three or four voices on a staff is far less common, but it does occur in important music, especially Bach, and we doubt any program handles such complex texture automatically at even a minimally acceptable level.)
					VDes	Horizontal position offset can be specified for all symbols on a staff; vertical position offset can be specified for all symbols on a staff and for staves.
3.6				x	VDes	All symbols can be specified as from an alternate source (perhaps editorial).
					Des	All symbols can be specified as from a second alternate source (perhaps editorial).
3.7			x		VDes	Noteheads, rests, slurs/ties, and dynamics can be specified as in parentheses or brackets, or as small.
					Des	All symbols can be specified as in parentheses or brackets, or as small.
3.8			x		VDes	Shape information can be provided for symbols such as tuplet brackets, octave-sign extenders, ties, and—most important—slurs. (Shapes of slurs can be very complex, with multiple inflection points (changes from concave to convex and vice-versa). Furthermore, there is no standard way of describing the shapes: the variety of Bezier curves used in PostScript is popular but not universal, and single curves of this type are not flexible enough to represent all slurs adequately. To prescribe such graphical information is beyond the scope of this document.)

No.	L	P	G	A	Pri.	Description
3.9			x		VDes	Any symbol, or at least notes, barlines, and text, can appear in color, with at least 16 colors supported. (This is mostly for pedagogical applications and for early music.)
3.10			x		Des	Any symbol, or at least key signatures, time signatures, and notes, can be invisible. (Some timpani parts of the classical period are best represented with invisible key signatures; time signatures can provide useful information for automatic, beaming, etc., even if they're not shown; making notes and perhaps other symbols invisible can be useful for pedagogic reasons. Tuplets are by far the most important invisible symbols, so they are treated separately: See Sec. 11.)
3.11				x	VDes	Information in the analytic domain is supported. (Examples include Kern's ability to specify key, as opposed to key signature (Huron 1997); Essen associative code's explicit phrase boundaries (Schaffrath 1997); Schenkerian indications of foreground, ornamentation, and so on.)
3.12			x		Req	The display of at least basic Schenkerian notation is supported, even if the representation does not put it in or does not support the analytic domain.
4. Notes and Chords						
4.1	x				Req	Both the starting times and durations for notes are given.
4.2	x				Req	The voice each note/chord belongs to is identified.
4.3	x				Req	The part each note/chord belongs to is identified.
4.4			x		Req	The staff each note/chord belongs to is identified.
4.5	x				Req	Basic durations from 128th to breve are supported.
					Des	Basic durations from 256th to longa are supported.
4.6	x				Req	0 to 2 augmentation dots are supported on durations of 32nd and longer.
					VDes	Additionally, 3 or 4 augmentation dots are supported on durations of 8th and longer. (Well-known pieces by Verdi, Sibelius, Hindemith, etc., use triple or even quadruple dots.)
4.7	x				Req	Supports both written and sounding pitches, expressed here in terms of ISO pitch notation, from C0 (9 ledger lines below bass staff, MIDI note number 12) to G#8 (12 ledger lines above treble staff, MIDI note number 116). Of course, simply giving written pitch and transposition (as defined above) is an acceptable way to do this; so is giving sounding pitch and transposition. (Written pitch involves a note "name" and octave, in some form. It is essentially an indication of the note's position on the staff, given a clef, possible alteration resulting from key signature and accidentals, and possible effect of an octave sign. The ISO notation was formerly known as the ASA (Acoustical Society of America) system. In this notation, the lowest full octave of the piano starting with C is octave 1, and each successive octave has the next higher number. So the lowest note on the piano is A0; the piano's lowest C, also the lowest sounding note on a five-string bass, is C1; middle C is C4, and the note just below it is B3. We do not mean in any way to specify that the representation must use ISO notation.) (C0 and G#8 are respectively the lowest and highest written notes we have ever seen.)
					Des	Supports basic pitches from MIDI note number 0 to MIDI note number 127.
4.8	x				Req	Supports the accidentals double-flat thru double-sharp, plus none.

No.	L	P	G	A	Pri.	Description
					Des	Supports triple-flats and triple-sharps. (While very rare, these have appeared in published music, and they could be useful in pedagogy or music-theory work.)
4.9	x				VDes	Supports microtonal accidentals, especially quarter-tones.
4.10			x		Req	Supports normal and small-sized accidentals.
					VDes	Also supports natural/flat and natural/sharp.
4.11			x		VDes	Supports the display of accidentals above or below the noteheads (for musica ficta or editorial accidentals; cf. general requirements for editorial notation, above). (Symbols in parentheses are covered in Section 3.)
4.12			x		VDes	Optionally force display of a note's effective accidental. (Such accidentals are often described as courtesy accidentals, but some atonal music uses them consistently.)
4.13		x			Des	Supports representing the MIDI note numbers for the pitch range specified in 4.7. (This is mostly to allow representing MIDI files without interpretation.)
4.14		x			Req	Supports a MIDI play resolution of 480 ticks per quarter or better.
					Des	Supports arbitrary MIDI play resolution.
4.15		x			Req	Supports performance durations of up to 100 whole notes.
					Des	Supports performance durations of up to 300 whole notes. (A Verdi opera contains a note with the longest duration we know of, about 250 measures of 4/4.)
4.16		x			Req	Supports MIDI On velocities from 1 to 127.
					Des	Supports the additional MIDI On velocity of 0 (meaning the note is silent, *not* indicating a note-off event).
4.17		x			Des	Provides for MIDI Off velocities from 1 to 127.
4.18			x	x	Req	Supports the following notehead shapes: normal (solid, half note, whole note, breve), invisible, x-shape, "harmonic" (hollow pseudo-diamond), chord slash.
					VDes	Supports these additional notehead shapes: Circle-x, hollow square, filled square, hollow diamond, filled diamond, half-note, triangle up, triangle down, arrow up, arrow down.
					Des	Supports additional shapes, including user-defined ones (cf. Roland 1997).
4.19			x		Des	Supports a special "artificial-harmonic note" with two heads. (Artificial harmonics can be done with harmonic noteheads, but it can be difficult to infer the actual pitch, especially with a chord of more than one normal head and/or harmonic head. An explicit representation for artificial harmonics is better.)
4.20	x				Des	Supports neumes. (These are used in Medieval music, especially for Gregorian chant. The set of music characters in Unicode includes about 10 neumes (Roland 1997; Unicode Consortium 2005). Note that neumes introduce questions about semantics, etc., since they represent multiple notes with one symbol.)
4.21			x		Req	Supports both normal and small noteheads.

No.	L	P	G	A	Pri.	Description
4.22	x				Req	Provides for the following note modifiers (articulation marks, ornaments, fingerings, etc.). • Digits 0 through 5 • Fermata • Normal accent (>) • Heavy accent • Staccato dot • Heavy accent with staccato • Wedge (marcato) • Tenuto • Trill with optional accidental • Mordent with optional accidental Inverted mordent with optional accidental Long inverted mordent with optional accidental Turn with one or two optional accidentals Plus sign (for left-hand pizzicato, brass stopped, etc.) Circle (for harmonics, brass open, etc.) Up-bow, down-bow Tremolo ("bowed") with specified number of slashes, 1-6 Heel and toe signs (for organ) Snap (Bartók) pizzicato With fingernails (for harp) Damp, damp all (for harp)
					VDes	Also provides for the following note modifiers: Arsis sign thesis sign double-tongue sign (..) triple-tongue sign (…) thumb position sign inverted (Wagner) turn with one or two optional accidentals *Sprechgesang* ("x" across stem) jazz effects: bend, flip, rip, smear, etc.
					Des	Also provides for the following note modifiers: Fermata lunga and poco a way to indicate fingering substitutions (i.e., starting a note with one finger and changing to another) nail pizzicato single-note/chord unmeasured tremolo (usually "z" across stem)

No.	L	P	G	A	Pri.	Description
4.23		x			VDes	Supports fine control of performance implications for note modifiers (to say how long a fermata is, etc.) (cf. Musedata "sound records," Finale articulation playback effects, Sibelius playback dictionary, and even the ancient Music Printer Plus codes).
4.24	x				Req	Supports chords of at least 10 notes.
					VDes	Supports chords of at least 16 notes.
					Des	Supports chords of at least 24 notes. (In two collections we checked totaling about 3300 scores, the densest chord has at most nine notes. But Scriabin put 24 noteheads on a single stem, and Cowell wrote tone clusters of over 50 notes.)
4.25			x		Req	Allows any chord that does not need diagonal stems.
					VDes	Also allows chords with diagonal stems for, at a minimum, augmented unisons. (Diagonal stems for an augmented unison are found as early as the Chopin Op. 10 Etudes, though they are quite rare in all 19th-century music we know of.)
					Des	Also allows chords for fully-notated tone clusters, i.e., clusters with every note individually represented.
4.26			x		VDes	Supports a special representation of tone clusters that gives just the lowest and highest notes.
					Des	For tone clusters in the special representation, allows specifying that the cluster includes only "black keys", only "white keys", all chromatic tones, or even notes in some microtonal scale.
4.27			x		VDes	Supports chords with notes on two adjacent staves of a part. (We are not aware of any clear-cut examples in which chords (as defined above) span more than two staves, but it seems likely that they exist, so allowing chords with notes on three adjacent staves is Des. Notice that the situation in Crumb's *Black Angels* is something different: there, stems connect notes not on multiple staves within a part, but in multiple parts, and therefore in different chords. Notice also that we do not specify that chords be represented explicitly, but if they are not, information applying to the chord as a whole, e.g., arpeggio signs, slurs, stem, will have to be stored with one or more notes, rather than where it really belongs.)
4.28			x		Req	Supports arpeggio and non-arpeggio signs.
					Des	Also supports arpeggio signs with arrows on bottom and (less important) top.
4.29				x	Req	Allows any note to have a stem, regardless of the notehead type.
4.30			x	x	Req	Allows stem direction to be specified.
4.31				x	VDes	Allows a single note to have both up and down stems.
4.32			x		Req	Allows length of note stems to be specified.
4.33				x	VDes	Allows any stemmed note to have an attached flag (useful for Schenkerian notation)
					Des	Allows a note both to be part of a beam group and to have a flag.
4.34			x		Des	Supports augmentation dots on the "wrong" side of the barline following the last note of a measure, e.g., when a 2/2 measure contains a half note followed by a dotted half. (This notation is fairly common in Baroque music; it also occurs occasionally in more recent composers, including Mozart, Beethoven, and Brahms.)
4.35				x	VDes	Supports at least 4 user-definable alphanumeric "tag" fields.

No.	L	P	G	A	Pri.	Description
					Des	At least 8 fields.
5. Grace Notes and Grace Chords Grace notes and chords have the same considerations as notes (the grace-note equivalents are **5.1** through **5.35**), except that for grace notes, **5.5**, **5.6**, **5.15**, **5.24**, and **5.28** are less demanding, and one additional requirement is provided (**5.36**).						
5.5	x				Req	Supports the following grace note durations: quarter, eighth, 16th, 32nd.
					VDes	Additionally supports 64th notes.
					Des	Additionally supports half notes.
5.6	x				Req	Supports up to zero augmentation dots (any system fulfills this "requirement").
5.15		x			Req	Supports performance durations of up to a half note.
					Des	Supports performance durations of up to a whole note.
5.24	x				VDes	Supports chords of at least 3 notes.
					Des	Supports chords of at least 5 notes.
5.28	x				VDes	Supports arpeggio and non-arpeggio signs.
					Des	Also supports arpeggio signs with arrows on bottom and (less important) top.
5.36	x				Req	For grace notes, allows stems to have an optional slash (for "real" grace notes, as opposed to appoggiaturas).
6. Rests						
6.1	x				Req	Supports the same requirements for specifying starting times and durations as for notes.
6.2	x				Req	Specifies voice identification in the same way as for notes.
6.3	x				Req	Specifies part identification in the same way as for notes.
6.4	x				Req	Specifies staff identification in the same way as for notes.
6.5	x				VDes	Provides for same basic durations as for notes, with this addition: supports multi-measure rests of up to 99 measures.
					Des	Supports multi-measure rests of up to 150 measures. (The longest multi-measure rest with an explicit number we have seen—in a percussion part of a Bruckner Symphony—is 128 measures.)
6.6			x		Req	Supports display of multi-measure rests using a horizontal bar on the middle staff line with the number of bars given above.
					Des	Supports use of the breve rest to represent two bars of rest, and the long rest to represent four bars of rest (with the number of bars indicated above the staff as well).
6.7	x				Req, VDes	Supports the same number of augmentation dots as for notes (see Item 4.6).
6.8			x		Req	Supports the same rest sizes as note sizes.
6.9	x				Req	Supports fermatas over a rest.
					Des	Supports the additional specification of these fermata types: lunga and poco; g.p. and l.p.

No.	L	P	G	A	Pri.	Description
7. Barlines, Measure Numbers, and Rehearsal Marks In this section, the term "barline" should be taken to include pseudobarlines in every case.						
7.1	x				Req	Supports the same starting time specification as for notes.
7.2	x				Req	Provides for these types of barlines: normal, double, final double, repeat-left double (||:), repeat-right double (:||), repeat-both double (:||:), dotted.
7.3	x				VDes	Supports storing where a repeat goes, when, and how often.
7.4			x		VDes	Double bars, repeats and dotted barlines can be set to act as pseudobarlines or not, on an individual basis.
7.5			x		VDes	Supports the independent placement of barlines in each group of staves (presumably for a part), i.e., not necessarily aligned with staves of other groups.
					Des	Supports the independent placement of barlines on each staff.
7.6			x		VDes	Supports Mensurstrichen, etc. (barlines that appear only between staves). (These are common in modern editions of Renaissance music; they have also been used in recent music, e.g., Stravinsky's Monumentum pro Gesualdo.)
7.7			x		Des	Allows specification of where in the score Mensurstrichen are used (rather than globally).
7.8			x		VDes	Supports "tic marks" thru the top of the staff as barlines. (Notes whose durations extend across Mensurstrichen are important: this might affect representation—though more likely of notes than of barlines.)
7.9	x				Req	Supports fermatas on barlines.
7.10	x				Req	Supports consecutive numbering of measures, subject to the fact that some barlines and repeat bars may not delimit measures. (The numbering of measures is a more subtle issue than it might appear. For instance, in published editions, measure numbers in a first ending are sometimes reused in a second or third ending, and sometimes they are not.)
7.11			x		Req	Provides a way to specify a starting measure number between 0 and 500.
					VDes	Provides a starting measure number up to 2200. (The greatest number of measures in any movement we know of is 2145.)
7.12	x				Des	Allows measure "numbers" that are not purely numeric, e.g., 29a, 29b. ("Numbers" like this sometimes appear in conjunction with endings, e.g., in the Wiener Urtext (ed. Badura-Skoda) and the Henle (ed. Gieseking) editions of Schubert piano music, and, apparently as a result of revisions, in Broadway musicals.)
7.13			x		VDes	Provides a way to specify which measure numbers are actually visible.
7.14			x		VDes	Supports specifying barline grouping—groups of staves that barlines extend vertically across—for the entire piece.
7.15	x				Req	Provides for rehearsal marks, attached to measures, of up to two characters.
					Des	Provides for rehearsal marks of up to 10 characters. (The longest rehearsal mark we know of is four characters unless one includes markings like "Reprise", "Chorus", etc., but these are still less than 10 characters.)
7.16			x		VDes	Supports optionally enclosing a rehearsal mark or measure number in a box or circle.

No.	L	P	G	A	Pri.	Description
8. Clefs						
8.1	x				Req	Supports the following clefs: C clef on any staff line, treble clef, bass clef, percussion clef, and no clef.
					Des	Additionally supports these clefs: French violin clef, baritone clef (F clef on the middle line), treble-tenor clef, bass clef with 8va sign below, Gregorian C and F clefs, "old" C clef, a second percussion clef. Cf. Roland (1997).
8.2	x				Req	Provides the ability to change clef on any staff at any point, even in the middle of a measure and regardless of the other staves.
8.3			x		Req	Provides for both normal and small-sized clefs.
8.4			x		Des	Supports simultaneous notes in two clefs on one staff. (While by no means common, this notational oddity is less rare than might be supposed: we have seen it in at least eight works of major composers, for example Debussy and Rachmaninoff (Byrd 1984, 1994).)
9. Key Signatures						
9.1	x				Req	Supports conventional key signatures with 0 to 7 sharps or flats.
					Des	Also supports key signatures with arbitrary placement of sharps or flats, including both at the same time. (These occur, e.g., in some works of Bartók.)
9.2			x		VDes	Supports optional naturals, to cancel a previous key signature or for scordatura.
9.3	x				Req	Supports the ability to change key signature on any staff at any point, even in the middle of a measure and regardless of the other staves.
10. Time Signatures						
10.1	x				Req	Supports time signatures with numerators of 1 to 99 and denominators of 1, 2, 4, 8, 16, and 32, plus no time signature at all.
					Des	Supports any integer denominator from 1 to 128. (We have seen denominators other than 1, 2, 4, 8, 16, and 32 only a very few times, and never in works of well-known composers, except for 64 and 128 in Crumb; those two values might also be useful in transcriptions of computer compositions or for pedagogy.)
10.2			x		VDes	Provides the ability to show the denominator as a note value, or to not show it at all. (The former is important in 20th-century music; the latter is important in early music.)
10.3	x				VDes	Supports compound time signatures and additive time signatures with up to three numerator components.
					Des	Supports additive time signatures with four numerator components and fractional (i.e., fraction in the numerator) time signatures. Examples of compound, additive, and fractional time signatures: 23 2+2+3 3-1/2 44 8 4
10.4			x		Des	Provides symbols for perfect and imperfect time, with perfect and imperfect prolation (for mensural notation).
10.5			x		Req	Supports C and "cut" symbols meaning, respectively, 4/4 and 2/2.
					VDes	Additionally supports C and cut with other meanings, e.g., 4/2 and 2/1.

No.	L	P	G	A	Pri.	Description
10.6					Req	Does not assume any kind of agreement between time signatures and the metric or rhythmic structure of the music.
10.7	x				Des	Supports, for any time signature, the description of metric structure independently of the visible time signature, e.g., (3+2+2)/8 for 7/8.
10.8	x				Req	Supports the ability to change time signature on any staff at any barline or pseudobarline, regardless of the other staves.
					Des	Supports the ability to change time signature at any point at all.
10.9			x		Des	Supports time signatures that extend across a group of staves.
11. Groups: Tuplets						
11.1	x				Req	Supports numerators ranging from 2 to 60.
					Des	Supports numerators ranging from 2 to 255.
11.2	x				Req	Supports denominators ranging from 1 to 20.
					Des	Supports denominators ranging from 1 to 255. (If the denominator is 1, presumably it would be displayed as the musical symbol instead of the number. The highest numerator in any tuplet we know of is 58, the largest denominator 14. But larger numbers might be useful for transcribing MIDI files and pedagogic purposes.)
11.3	x				Req	For the duration unit of the tuplet, supports undotted durations from half to 32nd.
11.4	x				VDes	Supports independent numerator and denominator duration units. (In the definition of duration unit above, this could handle Examples 1d—two half notes filling a measure of 3/4 and labeled "2"—and 1e—two quarter notes filling a measure of 5/8 and labeled "2". It still could not handle extreme cases like Example 1f.)
11.5	x				Des	Allows duration units to be dotted values. (This could handle Example 1f.)
11.6	x				Des	Supports duration units consisting of two (effectively tied) notes. (This affords an alternate way to handle Example 1e.)
11.7	x				Des	Supports duration units derived from tuplets.
11.8	x				Des	Supports additional denominator duration unit of whole, 64th, or 128th.
11.9			x		Req	Supports the display of any combination including none of numerator, denominator/duration, and "bracket" enclosing the notes and rests. Exception: showing the denominator/duration without the numerator is not required (nor is it desirable). ("Invisible" tuplets are commonplace, for example, in music in simple meter with long passages in triplets. Whether these tuplets have a visible manifestation or not, it must be possible to represent them; otherwise the rhythm is not represented correctly. On the other hand, brackets with neither numerator nor denominator visible is an important combination, if only because at least one major composer—Britten—used it repeatedly.)
11.10			x		VDes	Supports the specification of whether the denominator is displayed as a duration or a number.
11.11			x		VDes	Supports cross-system tuplets and tuplets across adjacent staves of a part.
11.12			x		VDes	Supports tuplets crossing barlines.

No.	L	P	G	A	Pri.	Description
11.13	x				Des	Supports nested tuplets.
12. Groups: Beams						
12.1			x		Req	Supports beams that connect any number of consecutive notes or chords, or any number of consecutive grace notes or grace chords, from 2 to 60, in a single voice.
					VDes	Supports from 1 to 150 notes/chords in a beam.
12.2			x		VDes	Supports beams connecting non-consecutive notes or chords in a single voice. (This is intended for bariolage for bowed strings, as usually notated in the Bach solo violin works and the last movement of Brahms' Symphony no. 4, and for the similar effect in harp music. Without this feature, such notation will require interleaving notes in multiple voices.)
12.3				x	Req	Supports beams that connect any number of notes or chords (including non-consecutive), from 2 to 60, in a single voice. (NB in analytic parts only; cf. Item 1.3. The ability to beam non-consecutive notes is important in Schenkerian notation.)
12.4				x	Req	Allows notes to be part of several beam groups at the same time.
12.5			x		VDes	Supports secondary beam breaks; fractional beams; feathered ("accelerando") beams; and cross-system beams.
12.6			x		VDes	Supports breaks/gaps in primary beams (useful especially in Schenkerian notation in which a beam spanning an entire system would obscure other symbols or otherwise add unnecessary clutter).
12.7			x		Des	Allows the intersection of a stem and beam to have a "V-notch" (used in Schenkerian notation).
12.8			x		Req	Supports beams crossing adjacent staves of a part.
12.9			x		Req	Supports beams crossing barlines.
12.10			x	x	Req	Supports center beams (i.e., stems going both up and down) on one staff. This also allows for the notation of the Schenkerian "unfolding."
12.11			x		VDes	Allows rests to be elements of non-grace beams.
12.12			x		Des	Provides control of fractional-beam direction (left or right).
12.13			x		Des	Provides for thin or normal beams.
12.14			x		Des	Supports control of beam height and angle. (This ordinarily determines stem lengths for all notes/chords in the beam.)
13. Groups: Octave Signs						
13.1			x		Req	Supports *ottava* and *ottava bassa,* affecting any number of notes, chords, grace notes, and grace chords on a staff.
					VDes	Also supports *15ma* and *15ma bassa,* affecting any number of notes, chords, grace notes, and grace chords on a staff.
13.2					Des	Octave signs can affect just certain voices on a staff, not all notes/grace notes on the staff. (We have seen this usage in works of Debussy and Tchaikovsky.)
13.3					Des	Octave signs can affect notes on more than one staff. (We have seen this usage in works of Ravel.)

No.	L	P	G	A	Pri.	Description
14. Tempo Markings "Instantaneous" tempo markings have two components, *text* and *metronome mark*; either or both may be present.						
14.1			x		Req	For the text component, supports any string of up 40 characters.
					Des	Supports a string of up to 100 characters.
14.2			x		Des	Supports embedding of a duration in a string.
14.3		x	x		Req	For the metronome mark component, supports specification in the form *dur* = *value*.
14.4		x	x		Req	Allows *dur* to be any duration, optionally with a single dot.
14.5		x	x		Des	Allows *dur* to be a series of tied notes.
14.6		x	x		Req	Allows *value* to be a number from 40 to 240 or a duration, optionally with a single dot. (Metronomes generally cover the range from M.M. 40 to 208. Values outside that range are very rare, but we have seen 20 to 640.)
					VDes	Allows numbers from 20 to 640 and a leading "ca." (or similar string).
14.7			x		Des	Allows optional parentheses around the metronome marking.
14.8			x		Des	Supports optional use of arrows to left and right (for the *dur* = *dur* form).
					Des	Allows for more of almost everything for the music of Elliott Carter; see Stone (1980) for discussion.
14.9		x	x		Req	Supports the "continuous" tempo markings "accelerando" and "ritardando," over any range of the score.
14.10			x		Req	Supports an optional trailing "poco a poco."
14.11			x		VDes	Supports the display of alternate wordings (e.g., "ritenuto").
14.12			x		VDes	Allows modifiers such as "molto," "poco," etc.
15. Text Strings and Lyrics						
15.1			x		Req	Supports display of arbitrary text strings in any available font and character set of up to 100 characters.
					Des	Support strings of up to 500 characters.
15.2			x		VDes	Provides Unicode support for all text (including names of parts). (NB: Over 200 music symbols are available in Unicode starting with version 3.1.)
15.3			x		VDes	Provides for an optional "extender" line of arbitrary extent after the text. (This can be useful for many purposes, such as melismatic lyrics, indications of the string to play, etc.)
15.4			x		VDes	Supports line breaks specified within the text.
15.5			x		Des	Supports mixing fonts, styles, sizes, baselines, and spacing with a single string.
15.6			x		VDes	Supports attaching text to any note, barline, slur/tie, repeat sign, or the page.
					Des	Supports attaching text to any symbol.
15.7			x		Des	Provides a way to indicate the language of the text.
15.8	x				Req	Provides a way to indicate whether a string is a lyric or not.
15.9	x				VDes	Provides a way to specify verse numbers of lyrics.

No.	L	P	G	A	Pri.	Description
16. Dynamics						
16.1		x			Req	Supports the "instantaneous" dynamics ppp, pp, p, mp, mf, f, ff, fff plus sf, sfz, rf, rfz, fp, sfp.
					VDes	Additionally supports pppp, ffff; relative dynamics piu p, piu f; and the markings piano possible, forte possible, sotto voce, mezza voce. (*Sotto voce* and *mezza voce* were used unequivocally as dynamics—not just nebulous expression marks—by Chopin, for one, in at least two Etudes.)
					Des	Additionally supports more *p's* and *f's*, and *mpp*, *mff*. (The most *p's* we have ever seen is 8; the most *f's*, is also 8.)
16.2		x			VDes	Allows a dynamic mark to affect two adjacent staves of a part, with or without a curly brace next to the dynamic.
					Des	Allows a dynamic mark to affect three or more adjacent staves.
16.3		x			Req	Supports "continuous" dynamics "*crescendo*" and "*diminuendo*" over any range of the score.
16.4			x		Req	Supports an optional trailing "*poco a poco*" on continuous dynamics.
16.5			x		Req	Supports display of continuous dynamics as hairpins (<, >).
16.6			x		VDes	Supports the display of alternate wordings (e.g., "*decrescendo*"), and the attaching of modifiers such as "*molto,*" "*poco,*" etc.
17. Slurs, Ties, Brackets, and Lines In this section, the word "slurs" should be taken to include phrase marks.						
17.1	x				Req	Allows a tie from a note to the next note in its voice if it is the same pitch, where "same pitch" is defined as generously as possible with respect to enharmonic notation, clef changes, etc.
17.2	x				Req	Allows a tie from a note to the right that is not attached to any following note, i.e., that goes into "thin air" or a following rest: this is for "laissez vibrer" situations.
17.3	x			x	Req	Allows slurs between any two notes/chords in the same voice.
					VDes	Also allows slurs between notes/chords in different voices.
17.4	x				Req	Supports nesting a tie and a slur.
					VDes	Supports nesting of a tie and up to two slurs.
					Des	Supports nesting of a tie and up to three slurs.
17.5	x				Req	Supports cross-system slurs/ties.
17.6			x		VDes	Supports slurs/ties crossing adjacent staves of a part.
					Des	Supports slurs/ties involving any staves of a part with three or more staves.
17.7			x	x	Req	Allows slurs and ties to be solid, dashed, or dotted.
17.8				x	Req	Supports the Schenkerian "hook" slur.

No.	L	P	G	A	Pri.	Description
17.9			x	x	VDes	Supports solid or dashed brackets—acting like slurs but consisting of straight horizontal or gently slanted lines with short vertical cutoffs—between any pair of notes (including in the same chord), e.g., to represent ligatures, or otherwise connect a group of notes.
17.10				x	VDes	Supports solid or dashed straight lines between any pair of notes.
17.11				x	VDes	Supports solid or dashed curved lines between any pair of notes.
17.12				x	Des	Lines (straight or curved) may have an arrow head at one or both ends.
17.13				x	Des	Supports braces between spanning any pair of notes (including notes in the same chord).
17.14				x	Des	Brackets, lines, and braces can connect any pair of symbols.
17.15			x		Req	Orientation for all shapes (curved upward or downward) can be specified.
17.16				x	VDes	Supports closed polygons enclosing any number of notes.
18. Staves and Staff Brackets						
18.1			x		Req	Supports staves with 1 or 5 lines.
					VDes	Additionally supports staves of 0 (for narrator, etc.), 4 or 6 lines.
					Des	Additionally supports staves of 2 or 3 lines.
18.2			x		VDes	Supports "cutout" staves, i.e., pieces of staves omitted completely instead of being filled with rests. (This device is common in 20th-century music, e.g., Crumb, Penderecki, late Stravinsky; it is also used in ossias in earlier music.)
18.3			x		VDes	Supports simultaneous large and small staves.
					Des	Supports simultaneous use of three staff sizes.
18.4			x		Des	Supports changing size of a staff at any point, even in the middle of a system (for incipits, etc.).
18.5			x		Des	Supports changing number of lines in a staff at any point, even in the middle of a system (for percussion, etc.).
18.6			x		Des	Supports staves with lines omitted.
18.7			x		Des	Supports staff lines in color.
18.8			x		Req	Supports curly brackets connecting the staves of a part, square brackets connecting any set of parts (but not overlapping other brackets).
18.9			x		VDes	Supports the choice of square brackets or no bracket instead of curly brackets connecting staves of a part.
18.10			x		VDes	Allows nested brackets across parts to two levels.
18.11			x		Des	Allows brackets to appear anywhere in a system, not just at its left end.
19. Annotation for Chords and Notes						
19.1	x			x	Req	Supports chord symbols for standard triads and 7th chords (e.g., "E", "G^{7}").
					VDes	Additionally supports 9th, 11th, and 13th chords with chromatic alterations of chord members (e.g., "$G^{7\#5}$"); added note chords (e.g., "$C^{\text{add }6}$"); SUS chords, and explicit bass notes (e.g., "G/F").

No.	L	P	G	A	Pri.	Description
19.2	x			x	Req	Supports figured bass symbols, stacked up to three high, consisting of numerals, optionally preceded by an accidental, and/or single accidentals aligned with a note; figures changing during a note (i.e., denoting a suspension or other linear motion).
					VDes	Additionally supports slashed numerals, leading and/or following +, "extender" lines, figures of arbitrary complexity. Cf. Hewlett (1997).
19.3				x	VDes	Supports scale-relative chord information ("roman numerals") associated with a note, including: • Scale degree and indication of major and minor triads (normally shown with upper- and lower-case roman numerals, respectively), plus N for Neapolitan • Indication of altered roots (normally shown with prefixed accidentals) • Indication of augmented, diminished, and half-diminished chords (normally shown with +, °, and slash-°, suffixes) • Information on inversions and more complex tertian chords equivalent to standard figured bass symbols 6, 6/4, 7, 6/5, 4/3, and 4/2, plus 9, 11, 13, with chromatic alterations of any chord member from the fifth up • Augmented sixth chords • Secondary function chords (We have not determined how to represent chords involved in a modulation. In a pivot-chord modulation, a chord is assumed to function simultaneously in two keys, e.g., C: IV and a: VI.)
19.4	x				VDes	Supports chord frames for six strings. (Explicit support for these is less essential because the graphics can be handled reasonably well with a font like Coda's Seville.)
					Des	Also supports chord frames for four or (for banjo, e.g.) five strings.
19.5				x	Req	Supports attachment of scale degree numbers to notes (^1 - ^8, plus optional accidentals).
19.6				x	Req	Supports attachment of figured bass symbols to notes (including between stems), above or below the staff.
					VDes	Supports dashes (of variable length if possible) between figured bass symbols to represent voice-leading motion over a note (e.g., 7----6).
20. Endings						
20.1	x				Req	Supports 1st and 2nd endings.
					VDes	Additionally supports 3rd endings.
					Des	Additionally supports 4th endings.
20.2	x				VDes	Supports combined 1st and 2nd endings.
20.3	x				Des	Supports combined 1st and 2nd, 2nd and 3rd, and 1st/2nd/3rd endings.
20.4	x				Req	Supports endings with a range of any length, indicated by a horizontal bracket and label.
21. Miscellaneous Graphic Elements						

No.	L	P	G	A	Pri.	Description
21.1	x				Req	Supports the following symbols: • Pedal down, pedal up • Caesura • Breath mark • Coda • Segno • D.C., D.S. • Repeat-beat sign (one slash alone; cf. Slash notehead) • Repeat signs with one and two slashes enclosed in dots • Glissando from one definite-pitched note to another • Fingered tremolo (Des: with optional fermata) • Schenkerian "interruption" (||)
					VDes	Additionally supports the following: • Harp pedal diagrams • Half pedal • Choice box ("frame notation") • Glissando from a definite-pitched note to a grace note • Indefinite slides preceding ("plop", "scoop") or following ("falloff", "doit") a note
					Des	Additionally supports the indication of prominent lines: Hauptstimme, Nebenstimme, unlabelled bold open angles, and cutoffs (bold close angles) for them; arrows; arbitrary straight and wavy lines, with and without arrowheads; choice of wiggly or straight lines on glissandi.
21.2		x			VDes	Supports performance implications for the above symbols where appropriate.
22. Miscellaneous Performance Elements						
22.1		x			Des	Supports attaching an arbitrary MIDI data stream to any note or even to any symbol (cf. NIFF); ideally it could be parameterized, as in the "MIDI macros" of the early Macintosh program ConcertWare.