Our Vision of the Future
----

* The entire repertory of classical music will be available free
  on the internet.

* The data can be downloaded either piecemeal or in bulk for any
  purpose the user desires.

* Possible uses include:.

Application	Technology Required
1. Searches and reference; checking data, expanding knowledge of repertory.	Display, search engines + indices
2. Historical and other studies; determining attribution.	Display, analytic tools
3. Music analysis; understanding methods of composition and modes of performance.	Display, possible AI applications
4. Printing of scores/parts for occasional, one-time use or for commercial publishing of both digiral and hardcopy editions,	Display, printing at verious sizes
5. Sound; listening to existing works and creating new ones.	Realistic artificial sound, AI applications

All of the above applications presuppose

* the existance of one or more massive databases of music
  in several formats

* with software that is able to access the data and convert
  it to a format desired by the user.

I suspect that there will never come a time when there is total convergence on an established set of formats for representing musical data.
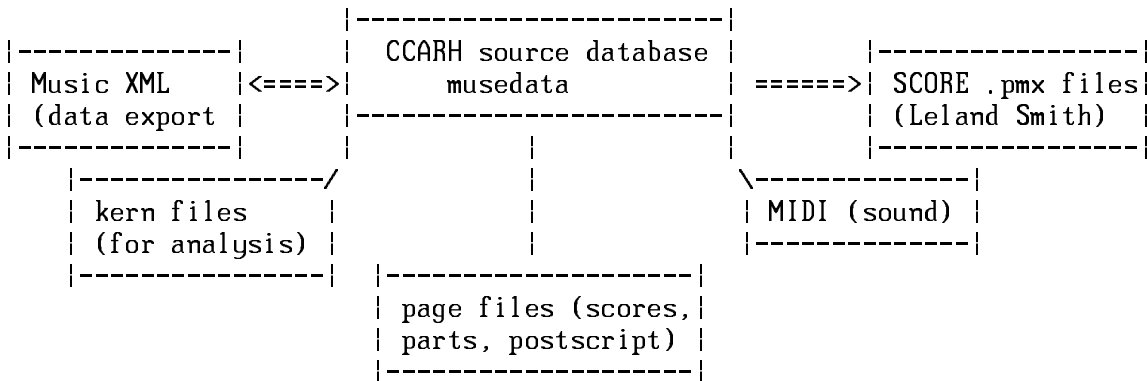
Reasons:

*   Music is complicated to represent
    -> and as a result, representations are often application dependent.

*   Representations that attempt to be universal
    -> are complex and can be difficult to work with.
    -> For specific applications it is often easier to develop
        a new representation than to try to use an existing one.

*   There will always be commercial opportunities in this field.
    -> Commercial enterprises will often have their own proprietary
        ways of representing musical data.

*   This is a field which is still in early development.
    -> We can expect that new knowledge will lead to new forms of
        representation and more powerful and easy-to-use
        software.

*   I expect AI will play a major role in the future design of
        databases and their formats.

Just as with formats, I suspect that there will never come a time when all available musical data will be found in one massive database.

*   I view musical databases as organic entities,
    -> which grow, flourish, and eventually beome obsolete;
    -> to be replaced by newer forms
    -> with the data, itself, being preserved and transformed
        as it moves from one database to the next.

*   At the present time, there are perhaps several dozen public or
        semi-public databases of music.

*   There may also be several times that number of private databases
        developed by individuals or small groups and used
        exclusively by them.
    -> (I am not including in this group individuals who have built
        private collections by simply downloadeding or purchasing
        sets of data).

How does the CCARH database fit into this picture?

```
                            |------------------------|
 |---------------|          |  CCARH source database |          |-----------------|
 | Music XML     |<====>|   |       musedata         |  ======>|  SCORE .pmx files|
 | (data export  |      |   |------------------------|   |      | (Leland Smith)  |
 |---------------|      |                 |              |       |-----------------|
     |---------------/  |                 |              \--------------|
     | kern files    |  |                 |              | MIDI (sound) |
     | (for analysis)|  |                 |              |--------------|
     |---------------|  |------------------------|
                        |  page files (scores,   |
                        |  parts, postscript)    |
                        |------------------------|
```

*   The Musedata representation is focussed on source data.

*   It is not driven by a specific application; nor
      is it particularly application friendly.

*   But it has the advantages of openness, adaptability, and simplicity.

*   Particular features of Musedata
      Flat files, organized in a Linux-type tree structure
      Accessable data, 7-bit ASCII code
      Data organized in columns. Column placement is an
        integral part of the format.
      Expandable
      Possible Outputs: display sound analysis hardcopy
      Other capabilities: searchability
        (using derivitive database structures)

*   Our style of working
      The incremental method
      Structured for building and adding on
      The incremental style of programming

*   Database structure

    The best way to understand the musedata format is to reveiw its
    historical development.

Fuga I.

a 4

Fuga I.

a 4

Fuga I.

a 4


```
253
846    2
Bach Gesells
chaft xiv
.
4   2
27   0   32   8
4   4   0   1   2
measure 1
rest      4
C4        4
D4        4
E4        4
F4        6
G4        1
F4        1
E4        4
A4        4
measure 2
D4        4
G4        6
A4        2
G4        2
F4        2
E4        2
F4        2
E4        2
D4        2
C4        2
D4        2
C4        2
B3        2
measure 3
A3        4
F#4       4
G4       12
F#4       2
E4        2
F#4       4
```

Fuga I.

a 4


```
253                 <--   number of records in this file
846   2             <--   BWV (work) number, Movement number
Bach Gesells        <--   source
chaft xiv
.
4 2                 <--   number of tracks; this track number
27 0 32 8           <--   number of measures; key; divisons per measure; divisons per quarter note
4 4 0 1 2           <--   time signature (two numbers, e.g., 4 4 );  clef (0 = treble clef);
measure 1           <--   beginning of first measure
rest    4           <--   rest;   number of divisions        (1 division = 32nd note, see above)
C4      4           <--   C4 is middle C;   number of divisions    (4 divisions = 8th note)
D4      4           <--   more notes, etc.
E4      4
F4      6
G4      1
F4      1
E4      4
A4      4
measure 2           <--   beginning of second measure
D4      4
G4      6           <--   notice that duration includes the tie.
A4      2
G4      2
F4      2
E4      2
F4      2
E4      2
D4      2
C4      2
D4      2
C4      2
B3      2
measure 3
A3      4
F#4     4
G4     12
F#4     2
E4      2
F#4     4
```

9.

Fuga I.

a 4

253	253
846 2	846 2
Bach Gesells	Bach Gesells
chaft xiv	chaft xiv
.	.
4 2	4 1
27 0 32 8	27 0 32 8
4 4 0 1 2	4 4 0 1 2
measure 1	measure 1
rest 4	rest 32
C4 4	measure 2
D4 4	rest 20
E4 4	G4 4
F4 6	A4 4
G4 1	B4 4
F4 1	measure 3
E4 4	C5 6
A4 4	D5 1
measure 2	C5 1
D4 4	B4 4
G4 6	E5 4
A4 2	A4 4
G4 2	D5 6
F4 2	E5 2
E4 2	D5 2
F4 2	C5 2
E4 2	measure 4
D4 2	B4 2
C4 2	G4 2
D4 2	A4 2
C4 2	B4 2
B3 2	C5 2
measure 3	B4 2
A3 4	C5 2
F#4 4	D5 2
G4 12	E5 2
F#4 2	D5 2
E4 2	E5 2
F#4 4	F#5 2

10.

Fuga I.
a 4


```
253                 253
846    2            846    2
Bach Gesells       Bach Gesells
chaft xiv          chaft xiv
.                  .
4 2                4 1               <-- track 1 of 4
27 0 32 8          27 0 32 8
4 4 0 1 2          4 4 0 1 2
measure 1          measure 1        <-- first measure
rest    4          rest   32        <-- 32 divisions = 1 whole measure
C4      4          measure 2
D4      4          rest   20        <-- note 20 divisions = 16 (half note) + 4 (8th note)
E4      4          G4      4
F4      6          A4      4
G4      1          B4      4
F4      1          measure 3
E4      4          C5      6        <-- C5 is the C one octave above middle C
A4      4          D5      1        <-- D5 etc.
measure 2          C5      1
D4      4          B4      4        <-- B4 is in the middle C octave
G4      6          E5      4
A4      2          A4      4
G4      2          D5      6
F4      2          E5      2
E4      2          D5      2
F4      2          C5      2
E4      2          measure 4
D4      2          B4      2
C4      2          G4      2
D4      2          A4      2
C4      2          B4      2
B3      2          C5      2
measure 3          B4      2
A3      4          C5      2
F#4     4          D5      2
G4     12          E5      2
F#4     2          D5      2
E4      2          E5      2
F#4     4          F#5     2        <-- # = sharp                      11.
```

Fuga I.

a 4

253	253	253
846 2	846 2	846 2
Bach Gesells	Bach Gesells	Bach Gesells
chaft xiv	chaft xiv	chaft xiv
.4 2	.4 1	.4 3
27 0 32 8	27 0 32 8	27 0 32 8
4 4 0 1 2	4 4 0 1 2	4 4 24 1 2
measure 1	measure 1	measure 1
rest 4	rest 32	rest 32
C4 4	measure 2	measure 2
D4 4	rest 20	rest 32
E4 4	G4 4	measure 3
F4 6	A4 4	rest 32
G4 1	B4 4	measure 4
F4 1	measure 3	rest 4
E4 4	C5 6	G3 4
A4 4	D5 1	A3 4
measure 2	C5 1	B3 4
D4 4	B4 4	C4 6
G4 6	E5 4	D4 1
A4 2	A4 4	C4 1
G4 2	D5 6	B3 4
F4 2	E5 2	E4 4
E4 2	D5 2	measure 5
F4 2	C5 2	A3 4
E4 2	measure 4	D4 6
D4 2	B4 2	E4 2

12.

Fuga I.

a 4

253	253	253	253
846 2	846 2	846 2	846 2
Bach Gesells	Bach Gesells	Bach Gesells	Bach Gesells
chaft xiv	chaft xiv	chaft xiv	chaft xiv

. 4 2	. 4 1	. 4 3	. 4 4
27 0 32 8	27 0 32 8	27 0 32 8	27 0 32 8
4 4 0 1 2	4 4 0 1 2	4 4 24 1 2	4 4 24 1 2
measure 1	measure 1	measure 1	measure 1
rest 4	rest 32	rest 32	rest 32
C4 4	measure 2	measure 2	measure 2
D4 4	rest 20	rest 32	rest 32
E4 4	G4 4	measure 3	measure 3
F4 6	A4 4	rest 32	rest 32
G4 1	B4 4	measure 4	measure 4
F4 1	measure 3	rest 4	rest 32
E4 4	C5 6	G3 4	measure 5
A4 4	D5 1	A3 4	rest 20
measure 2	C5 1	B3 4	C3 4
D4 4	B4 4	C4 6	D3 4
G4 6	E5 4	D4 1	E3 4
A4 2	A4 4	C4 1	measure 6
G4 2	D5 6	B3 4	F3 6
F4 2	E5 2	E4 4	G3 1
E4 2	D5 2	measure 5	F3 1
F4 2	C5 2	A3 4	E3 4
E4 2	measure 4	D4 6	A3 4
D4 2	B4 2	E4 2	D3 4

13.

Fuga I.

a 4

Three Problems:

1. Incompleteness

2. Inflexibility

3. Hard coding of tracks

Prelude I.

The Track Problem

Prelude I.

The Track Problem

Prelude I.

The Track Problem

Prelude VII.

The Track Problem

```
                    Conversion from stage1 to stage2
               ========================================
253                                  01/07/94 W Hewlett
846   2            <------->          WK#:846        MV#:2
Bach Gesells                          Bach Gesellschaft xiv
chaft xiv                             WTC I: Prelude and Fugue in C major
                                      Fugue
                                      Track 2
                                      1 2
                   concept of group --> Group memberships: sound openscr
4 2                   memberships     sound: part 2 of 4
                                      openscr: part 2 of 4
                                      &
27 0 32 8 \ become the "$" record    Initial conversion from stage 1 to stage 2
4 4 0 1 2 /                           &
measure 1 <-- removed                $  K:0   Q:8   T:1/1  C:4
rest  4                              rest    4       e
C4    4                              C4      4       e    u  [
D4    4                              D4      4       e    u  =
E4    4                              E4      4       e    u  ]
F4    6                              F4      6       e.   u  [
G4    1                              G4      1       t    u  =[[
F4    1                              F4      1       t    u  ]]]
E4    4                              E4      4       e    u  [
A4    4                              A4      4       e    u  ]
measure 2                            measure 2
D4    4                              D4      4       e    u  [
G4    6    <-- replaced with         G4      4-      e    u  ]      -
           8th tied to 16th          G4      2       s    u  [[
A4    2                              A4      2       s    u  ==
G4    2                              G4      2       s    u  ==
F4    2                              F4      2       s    u  ]]
E4    2                              E4      2       s    u  [[
F4    2                              F4      2       s    u  ==
E4    2                              E4      2       s    u  ==
D4    2                              D4      2       s    u  ]]
C4    2                              C4      2       s    u  [[
D4    2                              D4      2       s    u  ==
C4    2                              C4      2       s    u  ==
B3    2                              B3      2       s    u  ]]
measure 3                            measure 3
A3    4                              A3      4       e    u  [
F#4   4                              F#4     4       e #  u  ]
G4    12   <-- replaced with         G4      8-      q    u         -
           quarter tied to 8th       G4      4       e    u  [
F#4   2                              F#4     2       s    u  =[
E4    2                              E4      2       s    u  ]]
F#4   4                              F#4     4       e    u  [
D4    4                              D4      4       e    u  ]
measure 4                            measure 4
```

                                                                      19.

```
                 The column system of representation
         ------------------------------------------
I. Notes/Rests
    columns 1-8:      pitch/rest + duration (in time tics, right justified to column 8)
                         examples  C4 D4 E4 (C4 = middle C)
                                   Bf3 C4 D4 Ef4  (B-flat scale)   E4 F#4 G4  (E-minor scale)
    column  9:        tie flag   "-" = tie
    columns 10-14:    blanks (13 and 14 available for optional use)
    column  15:       track number (optional)
    column  16:       blank
    columns 17-22:    note description
            column 17: note type (letter code, Longa to 256th note) (also cue-size notes)
            column 18: dot flag  (. : ; !  one dot to quadruple dot)
            column 19: actual accidental flag (#,n,f, etc., 8 possibilities)
            columns 20-22: Two digits, separated by a colon (:) used for tuples
    column  23:       stem direction  u = up, d = down
    column  24:       staff number (" " = 1)
    column  25:       blank
    columns 26-31:    beams (up to six levels = 256th note)   [  = start beam
                         column 26 is for eighth beams,   =  = continue beam
                            27 for sixteenth,             ]  = end beam
                               ...                        /  = forward hook
                                                          \  = backward hook
    columns 32-43:    other notations, in no particular order or column
            Some examples:  Ties, Slurs, Tuples        Articulations and Accents
                            -  = tie                    >  = horizontal accent
                            (  = open slur1             .  = staccato
                            )  = close slur1            _  = legato
                            Ornaments                   Other Indications and Codes
                            t  = tr.                     F  = upright fermata
                            r  = turn                    p  = piano (pp, ppp, etc.)
                            ~  = wavy line (trill)       f  = forte (ff, fff, etc., fp)
                            M  = mordent                 m  = mezzo (mp, mf)
                            Technical Indications        Z  = sfz (also sf)
                            v  = up bow                  Zp = sfp
                            n  = down bow                R  = rfz
                     1,2,3,4,5  = fingering              +  = cautionary accidental

    columns 44-80:    text:  multiple lines of text set off by |
                            example:  Deck|See|Fast
```

II. Bar lines
    columns 1-7:   type of bar   e.g. measure = regular bar line
                                         mdouble = (light) double bar line
                                         mheavy2 = light-heavy double bar
    column 8:   empty
    columns 9-12:   optional bar number for this bar (left justified)
    columns 13-16:   blank or for optional use
    columns 17-80:   flags: some examples   F = fermata sign over bar line
                                  start-end# = start ending #
                                   stop-end# = stop ending #
                                   disc-end# = discontinue ending # line
                                         :| = repeat backward
                                         |: = repeat forward

III. Musical directions:
    column 1:   "*"
    columns 2-5:   blank
    columns 6-8:   optional forward offset (right justified)
    columns 9-14:   blank and optional information
    column 15:   track number (optional)
    column 16:   blank
    columns 17-18:   type of direction (two codes possible)
                          examples: D = left justified ASCII string
                                    (may be combined with types E,F,G,H,J)
                                    E = begin wedge
                                    F = end wedge (may be combined with types D,G,H,J)
                                    G = letter dynamics   (given in ASCII string)
                                    (may be combined with types D,E,F,H,J)
                                    H = begin dashes (after words)
                                    J = end dashes (may be combined with types D,G)
    column 19:   location flag (optional)
    column 20:   blank
    columns 21-23:   numerical parameter (e.g., wedge spread)
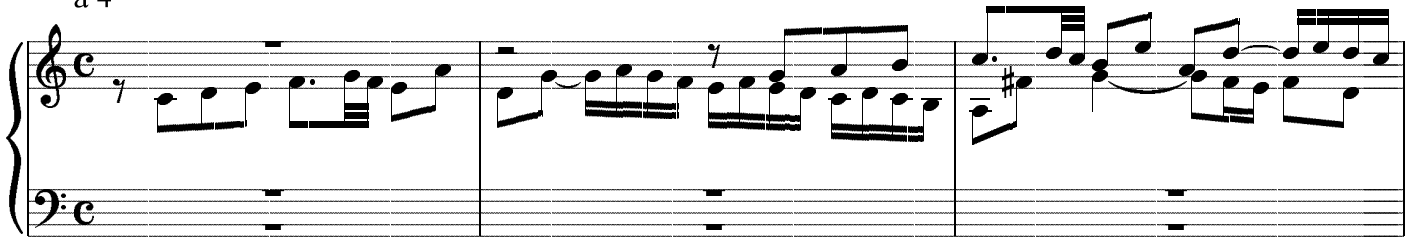   column 24:   staff number (' ' = 1)
    columns 25..:   ASCII word string   used in D,G
      Examples:
          1. cresc. - - - - - - - - ff   Starting record: DH   cresc.
                                             Ending record: JG   ff

          2. f <decreasing wedge> p   Starting record: GE 15 f
                                             Ending record: FG   0 p

          3. <increasing wedge> p   Starting record: E   0
                                             Ending record   FG 15 p

Fuga I.

a 4


```
$   K:0    Q:8    T:1/1    C1:4    C2:22    |Musical attributes: key, divspq, time, clefs
rest  32          1        1                |(track 1) rest       duration, track, staff
back  32                                    |backup command (to start of measure)
rest   4          2 e      1                |(track 2) note data: pitch, duration, track,
C4     4          2 e      d1 [             |            type of note, extension dot(s)
D4     4          2 e      d1 =             |            stem direction, staff number
E4     4          2 e      d1 ]             |            beaming information
F4     6          2 e.     d1 [             |
G4     1          2 t      d1 =[[           |
F4     1          2 t      d1 ]]]           |
E4     4          2 e      d1 [             |
A4     4          2 e      d1 ]             |
back  32                                    |backup command
rest  32          3        2                |(track 3) rest (staff 2)
back  32                                    |backup command
rest  32          4        2                |(track 4) rest (staff 2)
measure 2                                   |Bar line
rest  16          1 h      1                |(track 1) half note rest
rest   4          1 e      1                |          eight rest
G4     4          1 e      u1 [             |
A4     4          1 e      u1 =             |
B4     4          1 e      u1 ]             |
back  32                                    |backup command (to start of measure)
D4     4          2 e      d1 [             |
G4     4-         2 e      d1 ]        -    |                   tie
G4     2          2 s      d1 [[            |
A4     2          2 s      d1 ==            |
G4     2          2 s      d1 ==            |
F4     2          2 s      d1 ]]            |
E4     2          2 s      d1 [[            |
F4     2          2 s      d1 ==            |
E4     2          2 s      d1 ==            |
D4     2          2 s      d1 ]]            |
C4     2          2 s      d1 [[            |
D4     2          2 s      d1 ==            |
C4     2          2 s      d1 ==            |
B3     2          2 s      d1 ]]            |
back  32                                    |backup command
rest  32          3        2                |(track 3) rest (staff 2)              22.
back  32                                    |backup command
rest  32          4        2                |(track 4) rest (staff 2)
measure 3                                   |Bar line
```

Fuga I. (system 2, bars 4, 5, and 6)


```
measure 4                             |Bar line
B4       2      1 s     u1 [[         |(track 1)  > four 16ths
G4       2      1 s     u1 ==         |           > beamed
A4       2      1 s     u1 ==         |           > together
B4       2      1 s     u1 ]]         |           >
C5       2      1 s     u1 [[         |           etc.
B4       2      1 s     u1 ==         |
C5       2      1 s     u1 ==         |
D5       2      1 s     u1 ]]         |
E5       2      1 s     u1 [[         |
D5       2      1 s     u1 ==         |
E5       2      1 s     u1 ==         |
F#5      2      1 s #   u1 ]]         |accidental in column 19 (sharp)
G5       4      1 e     u1 [          |           > two eights
B4       4      1 e     u1 ]          |           > beamed together
back    32                           |backup command
G4       4      2 e     d1 [          |(track 2)  > four 16ths
F4       4      2 e     d1 =          |           > beamed
E4       4      2 e     d1 =          |           > together
D4       4      2 e     d1 ]          |           >
C4       4      2 e     d1            |
rest     4      2 e      1            |
rest     4      2 e      1            |
G4       4-     2 e     d1        -   |
back    32                           |backup command
rest     4      3 e      2            |(track 3) eight rest
G3       4      3 e     u2 [          |          note data: pitch, duration, track,
A3       4      3 e     u2 =          |             type of note, stem direction
B3       4      3 e     u2 ]          |             staff number = 2
C4       6      3 e.    u2 [          |             beaming information
D4       1      3 t     u2 =[[        |
C4       1      3 t     u2 ]]]        |
B3       4      3 e     u2 [          |
E4       4      3 e     u2 ]          |
back    32                           |backup command
rest    32      4        2            |(track 4) rest (staff 2)
measure 5                            |Bar line
```

Print Suggestions

I. What is the purpose?

* It is possible using an automated process to produce reasonably good
        musical output from musedata files.

* The output can be improved by adding print suggestions to the files.

* Compactness is a desirable quality in musical output, but making
        something compact can introduce spacing and overstrike problems.

* It turns out that many of these clashes can be "fixed" if their vertical
        and/or horizontal positions can be adjusted slightly.

II. How does this work?

* For items represented in columns 32 to 43 of a note/rest record,
        what is needed is the column number of the item.

* If, for example, you have a fermata (F) indicated in column 35,
        the suggestion C35:y-10 will move the fermata up 10 units

* A print suggustion record must follow directly after the data record
        and must have the letter "P" in column 1. In the example above
        the print suggestion record would look like this:
            P   C35:y-10

+ It is possible to have more than one suggestion in a P record

III. The print suggestion feature has many other uses:

   1. Position of slurs (over notes, under notes)
   2. Orientation of ties (overhand, underhand)
   3. Suggestions for representing beamed notes with repeaters
   4. Suggestions for musical directions
        for example: c<#> where # is between 0 and 15
            0 = print always (default)
            bit 0 set = print in parts   bit 2 set = print if top part in score
            bit 1 set = print in score   bit 3 set = print if bottom part in score
   5. Suggestions for treatment of whole measures
        for example: C1:]   = use system justification to force
                                 this bar line to the end of a system.
        This is used to adjust the layout, e.g., For page turns in musical parts.

There is also a long list of general print suggestions.

The field designator for a general print suggestion is C0:
Examples of some suggestion codes:

```
  d<#>  = default height for time words and other musical designations.
  f<#>  = default font for musical directions in "*" records
  h<#>  = alter the minimum allowed space between notes
          # = percentage of default size  (100 = default)
  k<#>  = various operational flags (defaults are 0)
    bit 0: (for two or more tracks)
       0 = allow overstrike when there is a dot-difference  1 = do not overstrike
    bit 1: (for printing new key signatures)
       0 = don't print a new key signature if it is the same as the previous one
       1 = always print a key signature even when it hasn't changed
    bit 2: (chords with white and black notes)
       0 = don't allow a mixture of white and black notes in chords
       1 = allow mixture of white and black notes
    bit 3: (for suppression of the key signature)
       0 = normal  1 = suppress printing of the key signature (for timpani parts, etc.)
    bit 4: (for assigning editorial slurs)
       0 = normal (no assignment) 1 = "{ }" and "z x" combinations are editorial slurs
    bit 5: (for printing new clef signs)
       0 = normal (large clefs, only at start of measure)  1 = always use large clefs
    bit 6: (for printing sfortzandos)
       0 = normal (Z = sf or sfz)     1 = abbriviated (Z = fz)
    bit 7: (for printing unisons in chords)
       0 = normal (side-by-side)  1 = over-strike
  p<#>  = minimum distance between notes (expressed as percent of the default).
  q<#>  = duration which is assigned the minimum distance
          0 = recompute default (from this point onward)
          1 = whole notes  . . .  8 = eighth notes, 16 = sixteenth notes, etc
  s<#>  = space between grand staffs measured in multiples of ledger lines times 10
          (e.g. 100 = 10 ledger lines).
  t<#>  = global tuplet placement
          # = 0: use default
          # = 1: place tuplet near note heads
          # = 2: place tuplet near note stems (beams)
          # = 3: place all tuplets above notes
          # = 4: place all tuplets below notes
  v<#>  = location of text below music  # = number of scale steps (x 10)
          (default approx. 150)
```

All print suggestions in a musedata file are optional and can be overridden
by a music typesetting program.

T H E   I - F I L E   F O R M A T
===

* I-Files come in two types: linear and page spacific.
  -> A linear file represents one voice or part in its entirety.
  -> A page spacific file represents one or more parts (a musical score)
       on a single page. We call this a music page file.

* A music page file consists of a list of glyphs (characters from a music font)
     and their location on the page.
  -> A musical note, for example, consists of a note head, one or more pieces of
       stem, possibly a flag, an accidental, and some leger lines, etc., etc.

* From a technical point of view, the order of the glyphs in the list
       shouldn't matter (as long as the x,y position is specified).
  -> But for purposes of editing, some choices of ordering are better than others.
  -> Let us imagine, for example, that we want to move the musical note
       to a new (x,y) position on the page.
  -> In order for the note to stay together, we must move all parts of
       it by the same amount.
  -> It makes sense, therefore, to think of the note as a single unit, even
       though it has several parts.

* To implement this concept, we define something we call an OBJECT.
  -> The object is our basic unit of musical notation.
  -> The object may consist of several glyphs, as in the case of a note.
  -> The glyphs, themselves, we call SUB-OBJECTS, because they are members
       of the thing we call an object.
  -> The position of sub-objects is specified in relation to the object to
       which they belong.
  -> This way, if we want to move a note, we simply move the object, and
       all the parts of it (the sub-objects) will move together.

* All basic units of musical notation (objects) are attached to a STAFF LINE.
  -> It therefore makes sense to specify the location of an object in relation
       to the staff line to which it belongs.
  -> This way, if we move a staff line, all notes and other notation connected
       with that line will move together.

* All staff lines belong to a SYSTEM.
  -> A system may have one or more staff lines associated with it.
  -> As before, it makes sense to specify the location of a staff line in
       relation to the system to which it belongs.

* Systems represent the highest level in the location hierarchy on the page.
  -> The location of a system is therefore specified by absolute (x,y)
       co-ordinates on the page.

* To summarize, let us take the example of a sharp (#) attached
     to a note. The absolute (x,y) of that sharp will be:

          The absolute location of the system (x,y), plus
          the (dx,dy) offset to the staff line in the system, plus
          the (dx,dy) offset to the note on the staff line, plus
          the (dx,dy) offset to the sharp (#) from the note.

* Much of musical notation can be represented in the manner described
   thus far.
   -> Notes, rests, text attached to a note, musical ornaments, articulations
   -> Dynamics, musical directions, bar lines,
   -> Time signatures, key signatures, clefs,
   -> basically anything that stands by itself on the musical page.

* There is a class of things, however, that should not be represented as objects,
   because the position of these things depends on the position of more than
   one object. Items in this class include:
   -> ties, beams, slurs, tuplets (and their brackets),
   -> endings, long trills, transposition lines,
   -> dynamic wedges, and dashes associated with changes in dynamics, tempo, etc.

* We call this class of things, SUPER-OBJECTS, because their position (and shape)
   depend on more than one object.

* In actual fact, most super-objects are printed using glyphs from the music fonts.

* However, they are not represented in the music page file in this manner,
   -> because if we move a object which has a super-object associated with it,
      the position and the shape of that super-object (and therefore the
      glyphs that comprise it) will change.
                     h
* A music page file consists of list of variable length records. The order of
   the records is an integral part of the representation. There are 12 types of
   records. The character in column one of a record identifies its type.

         Record Type   Identifier (column 1)
        -------------------------------   ----------------------
         Header Record   Z
         Page oriented text   X
    ->   System (page oriented)   S
    ->   Staff Line   L
    ->   Objects   J
    ->   Sub-objects   K
         Text (form of sub-object)   T
         Words (form of sub-object)   W
         Attribute (form for sub-object) A
    ->   Super-objects   H
         End of Music Line   E
         System Bar   B

* The basic location hierarchy is: System - Line - Object - Sub-object.
   -> The order of records in the music page file reflects this hierarchy.
   -> A system record remains active in the list until a new system record is
      encountered.
   -> A staff line record remains active until an End of Music Line record is
      encountered.
   -> Objects on a staff line are listed in the time order (left to right)
      in which they occur on the staff line.
   -> The sub-objects that belong to an object are listed directly below the
      object.
   -> System bars are listed after the last staff line of a system
      has been fully represented.
   -> Page oriented text should be listed before or between systems.
   -> Header records belong at the beginning of the file.

Haydn Op64, No.1, 1st movement
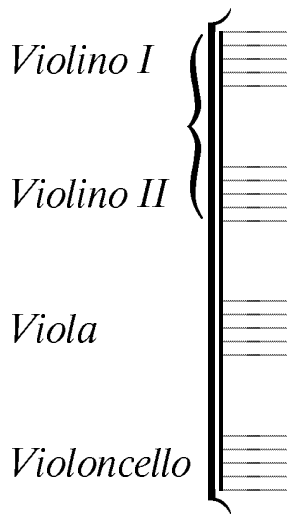## Allegro moderato

Violino I

Violino II

Viola

Violoncello

x and y Units are in "dots"   300 dots = 1 inch.

Origin x,y = <0,0> is top left of page   x increases to the right; y increases going down

Data types (column 1): S=system, L=line, J=Object, K=Sub-object, W=word (a sub-object),
  E=end of line, B=barline
  All of this to indicate the position of the glyphs that are the heart of the
  music printing system.

```
X 21                             Notesize: 21 dots between staff lines
S 0 500 240 105 714 4 "[{..}..]"  System:  x=500 y=240 is location on page.  length=105
L 0 0 0 0 0 * 0 21 0             Line:    y position = 0 is offset reletive to the system
J D 0 -326 63 1 6913 0 0         J (Object)  x,y position (-326,63) offset relative to the line
W 0 0 39 Violino I               W (Word)    x,y postion (0,0) offset relative to the object
E *                              E indicates End of line
L 210 0 0 0 0 * 0 21 0           Line:    y position = 210 is offset relative to the system
J D 0 -326 63 1 6913 0 0
W 0 0 39 Violino II
E *
L 420 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Viola
E *
L 630 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Violoncello
E *
B 1 1 0
```

Types of Objects: N=note, R=rest, D=directive, K=key, C=clef, T=time signature,
                   B=barline, G=grace note, Q=cue note, F=figured harmony, S=symbol

Sub-objects are music glyphs
A Word (W) is a form of sub-object

```
X 21
S 0 500 240 1750 714 4 "[{..}..]"    length of system is now 1750 dots    height is 714 dots
L 0 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Violino I
J C 4 7 63 2 6913 0 0     J Object is a Clef   x,y position (7,63) is offset relative to the line
K 0 0 33                  K is Sub-Object (a glyph)  The treble clef is made up of
K 0 0 34                      two glyphs, 33 and 34
E *
L 210 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Violino II
J C 4 7 63 2 6913 0 0
K 0 0 33
K 0 0 34
E *
L 420 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Viola
J C 13 7 42 35 6913 0 0  J Object is a Clef   x,y position (7,42) is offset relative to the line
                             In this case, the Object is the glyph (glyph number is 35).
E *
L 630 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Violoncello
J C 22 7 21 36 6913 0 0
E *
B 1 1 0
```

```
S 0 500 240 1750 714 4 "[{..}..]"
L 0 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Violino I
J C 4 7 63 2 6913 0 0
K 0 0 33
K 0 0 34
J T 0 85 63 1 6913 0 0          J Object is a Time Signature.  Offset from line is (85,63)
K 0 0 38                          Sub-Object: glyph = 38 (cut time)
J N 7 157 136 10 1 0 0          J Object is a Note           Offset from line is (1575,136)
K 1 -31 45                        Lots of Sub-Objects (glyphs) here
K 1 -10 45                          45 = leger line   Offset from the Object is <1,-10>
K 0 0 43                             43 = note head (black note)
K 0 -10 45
K 2 -10 45
K 0 42 108                          108 = piano dynamic (offset <0,42> measured from the Object
K 0 0 59                              59 = up stem (length = two staff line widths (42 dots)
K 0 -42 59
K 0 -52 59
A D 12 48                       This is duration information, used for analysis not display
E *
L 210 0 0 0 0 * 0 21 0
J D 0 -326 63 1 6913 0 0
W 0 0 39 Violino II
J C 4 7 63 2 6913 0 0
K 0 0 33
K 0 0 34
J T 0 85 63 1 6913 0 0
K 0 0 38
J R 7 157 42 2 1 0 0
K 0 0 48
A D 12 48
E *
```

```
S 0 500 240 1750 714 4 "[{..}..]"
L 0 0 0 0 0 * 0 21 0
    (lines removed to create space)
J B 1 230 1 82 6913 576 0          Object: Barline (bar 1) Glyph-82
J N 8 262 105 7 1 0 0              Object: Next note, half-note
K 1 0 45                             lots of Sub-objects (glyhgs) to display this
K 0 0 42
K -1 0 45
K 5 0 45
K 0 0 59
K 0 -32 59
A D 24 48
J N 6 371 84 3 3457 1152 2 1 2     This note object has 2 super-objects (nos. 1 and 2)
K 0 0 43                              Super-object 1 is a slur; super-object 2 is a beam.
K 36 -9 44                            The stems will be added whem the beam is drawn.
A D 9 48
J N 5 429 105 5 4753 432 2 2 1      This note object has 2 super-objects (nos. 1 and 2)
K 1 0 45                              The stems will be added whem the beam is drawn.
K 0 0 43
K -1 0 45
K 5 0 45
A D 3 48
H 2 B 72 4 112 2 2 53              H Super-object number 2: This one a beam
H 1 S 12 0 0 0 0 0 0 0 0           H Super-object number 1: This one a slur
J N 6 469 73 3 5185 144 2 3 4
K 0 0 43
K 38 0 44
A D 9 48
J N 5 527 94 2 6481 432 2 4 3
K 0 0 43
A D 3 48
H 4 B 61 4 112 2 2 53              (Two more super-objects)
H 3 S 12 0 0 0 0 0 0 0 0
J B 2 568 1 82 6913 144 0          Object: Barline (bar 2) Glyph-82
E *
L 210 0 0 0 0 * 0 21 0             Next staff lines
J D 0 -326 63 1 6913 0 0
etc.
```
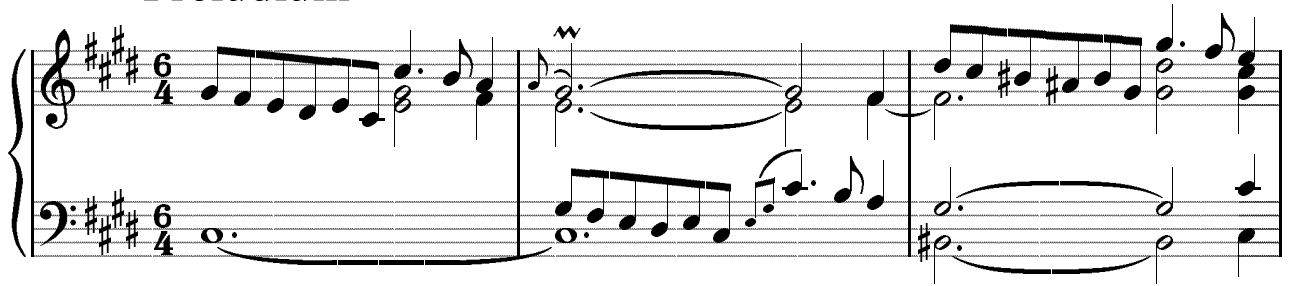
35.

Preludium


```
X 46 575 120 Praeludium IV.           Text Record: Title
S 0 500 240 1750 226 1 "({.})"        System No. 1
L 0 0 0 0 0 * 170 14                     Lines No. 1 & 2
J D 0 -300 42 1 6913 0 0
W 0 0 39
J C 4 5 42 2 6913 0 0                  Object: Clef
K 0 0 33
K 0 0 34
J C 22 5 1014 36 6913 0 0             Object: 2nd Clef
J K 4 57 0 4 6913 0 0                  Object: Key signature
K 0 0 63
K 15 21 63
K 30 -7 63
K 45 14 63
J K 4 57 1000 4 6913 0 0              Object: 2nd Key signature
K 0 14 63
K 15 35 63
K 30 7 63
K 45 28 63
J T 604 142 0 2 6913 0 0              Object: Time signature
K -10 28 77
K -10 56 75
J T 604 142 1000 2 6913 0 0           Object: 2nd Time signature
K -10 28 77
K -10 56 75
J N 6 181 42 2 1 0 1 1                Object: Note G#4
K 0 0 43                                 Sub-object: note head
A D 1 8                                  Attribute: duration 1/8th
J N 9 181 1035 3 1 0 1 2              Object: Note C3 (2nd line)
K 0 0 41                                 Sub-object: note head
K 32 0 44                                Sub-object: dot
A D 12 8                                 Attribute: duration 12 8ths
J N 6 222 49 2 577 288 1 1            Object: Note F#4
K 0 0 43
A D 1 8
J N 6 263 56 2 1153 288 1 1           Object: Note E4
K 0 0 43
A D 1 8
J N 6 304 63 2 1729 288 1 1           Object: Note D#4
K 0 0 43
A D 1 8
J N 6 345 56 2 2305 288 1 1           Object: Note E4
K 0 0 43
A D 1 8
J N 6 386 70 5 2881 288 1 1           Object: Note C#4
K 1 0 45
K 0 0 43
K -1 0 45
K 2 0 45
A D 1 8
H 1 B 45 2 108 6 2 1 1 1 1 3          Super Object: Beam (six notes)
J N 7 427 21 5 3457 288 0             Object: Note C#5
```

Glyph Numbers for the Music Font
=====================================

4.1   The glyph numbers appear mainly as the fourth field of sub-ojbects.
      Occasionally one finds a glyph number in field 6 of an ojbect. The
      table below shows the shapes assigned to the various glyph numbers.

1. large clefs
   ----
      33 treble (top)
      34 treble (bottom)
      35 C-clef
      36 bass

2. large time signatures
   ----
      37 common time
      38 alle breve time

3. full-size note heads
   ----
      39 longa
      40 breve
      41 whole
      42 half
      43 quarter

4. full-size dot, leger line
   ----
      44 dot
      45 leger line

5. full-size rests
   ----
      46 whole
      47 half
      48 quarter
      49 eighth
      50 add-eighth

6. full size flags
   ----
      51 short up-eighth
      52 short down-eighth
      53 up-eighth
      54 down-eighth
      55 up-sixteenth
      56 down-sixteenth
      57 up-add-flag
      58 down-add-flag

7. full size stems
   ----
      59 full length up
      60 full length down
      61 notesize up
      62 notesize down

8. accidentals
   ----
      63 sharp
      64 natural
      65 flat
      66 double sharp

9. editorial brackets
   ----
      67 square left
      68 square right
      69 round left
      70 round right

10. big numbers
    ----
      71 0
      72 1
      73 2
      74 3
      75 4
      76 5
      77 6
      78 7
      79 8
      80 9

11. staff lines
    ----
      81 full-size staff

12. vertical lines
    ----
      82 full length bar
      83 quarter length bar
      84 full leng. thick bar
      85 qtr. length thick bar
      86 full leng. dotted bar
      87 thick vertical top
      88 thick vertical bottom

13. horizontal lines
    ----
      89 begin/end hook
      90 solid line (30 dots)
      91 dash line (30 dots)
      92 heavy line (30 dots)

14. articulations (10)
    ----
      93 horizontal accent
      94 ^ accent
      95 ∨ accent
      96 staccato dot
      97 ∨ stricht
      98 ^ stricht
      99 - legato
      100 , breath
      101 fermata up
      102 fermata down

15. repetition (5)
    ----
      106 signet sign
      107 circle + cross sign

16. dynamics (6)
    ----
      108 p
      109 m
      110 f
      111 s
      112 z
      113 r

37.

```
                    Final summary
              -------------------------
```

*   Early databases in music consisted mainly of the captured keystrokes
         from the data entry portion of music display programs.
     -> DARMS, Plaine-and-easie, and SCORE .mus files are three examples
         that come to mind.

*   The next step up was the development of data formats that were used
         directly by music printing programs.
     -> The SCORE .pmx files and the Finale Enigna files are two that fall
         in this category.

*   The next step up is what I would call application independent databases.
     -> Our musedata format, the humdrum format and the MEI format are three
         examples.

*   What sets application independent databases apart from the others is
        that they seek to represent the various elements of music indepentent
        of whatever software might be written to use the data.

*   Our goal in setting up the format for musedata was to provide a way to
         encode vast quantities of historial musical literature --
     -> literature that is now fixed in time and doesn't for the most part
         change, and
     -> liturature for which the coding could be done once and not have to be
         redone every time a new edition is created.