

# An Introduction to SCORE

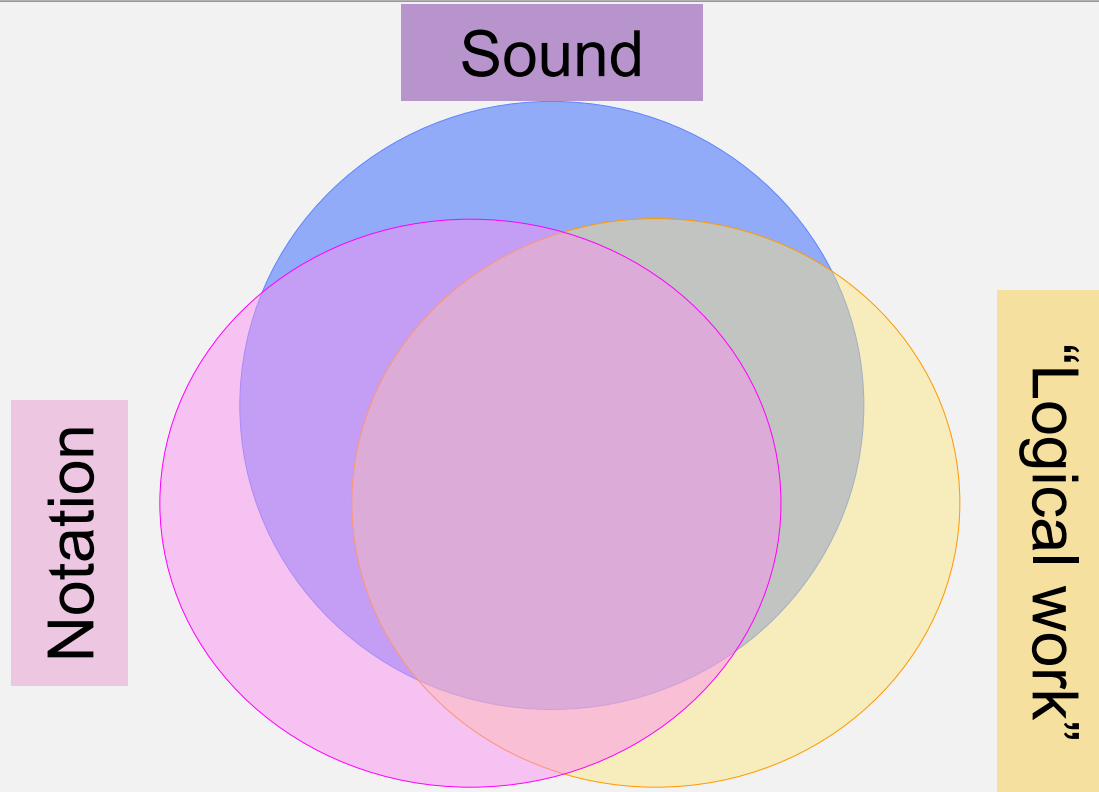
---

MUSIC 253/CS 275A

STANFORD UNIVERSITY

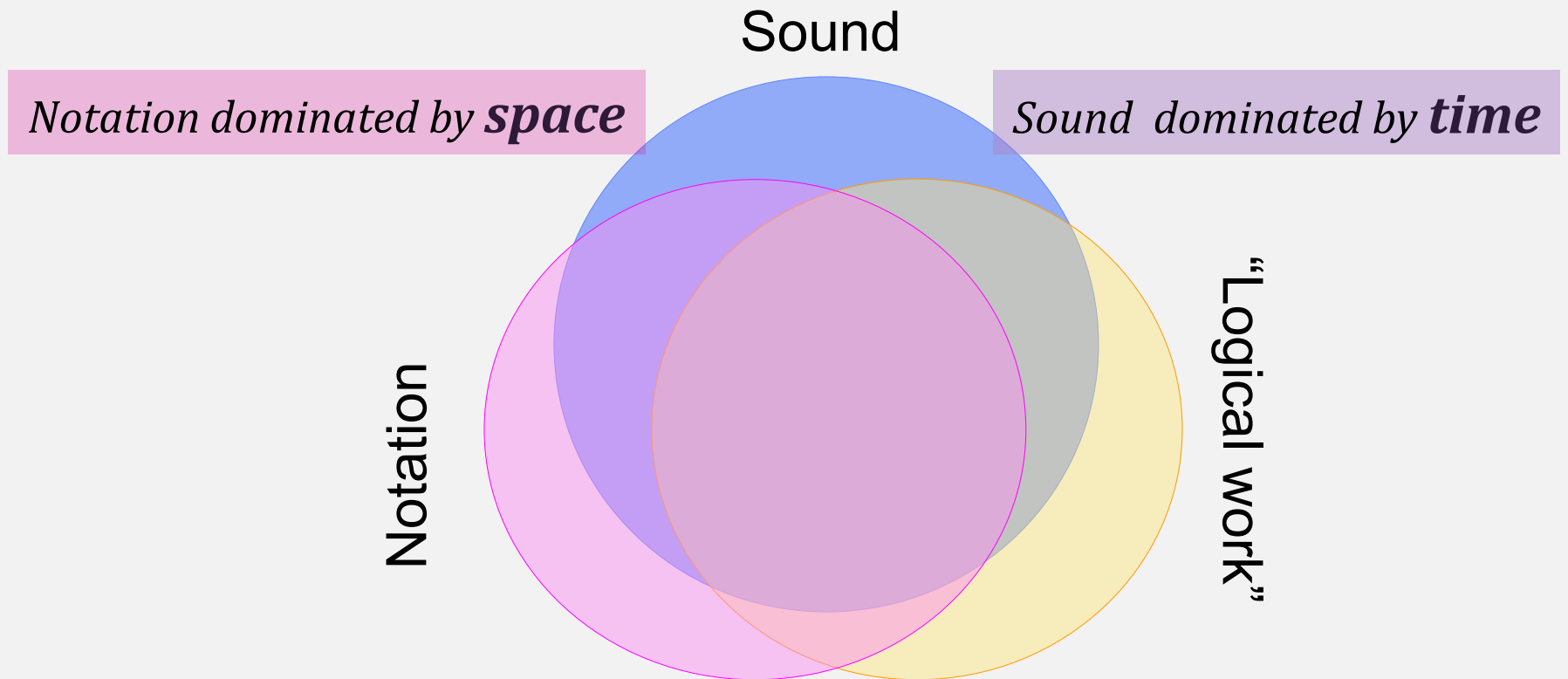
# The Graphics (*Notation*) Domain

---



# The Graphics Domain

---



# The Graphics Domain: Basic Principles

---

## PITCH

Height on a staff

### **Pitch inflection**

Marks (#, B, etc)

### **Pitch articulation**

Marks (*staccato*, *tr*)

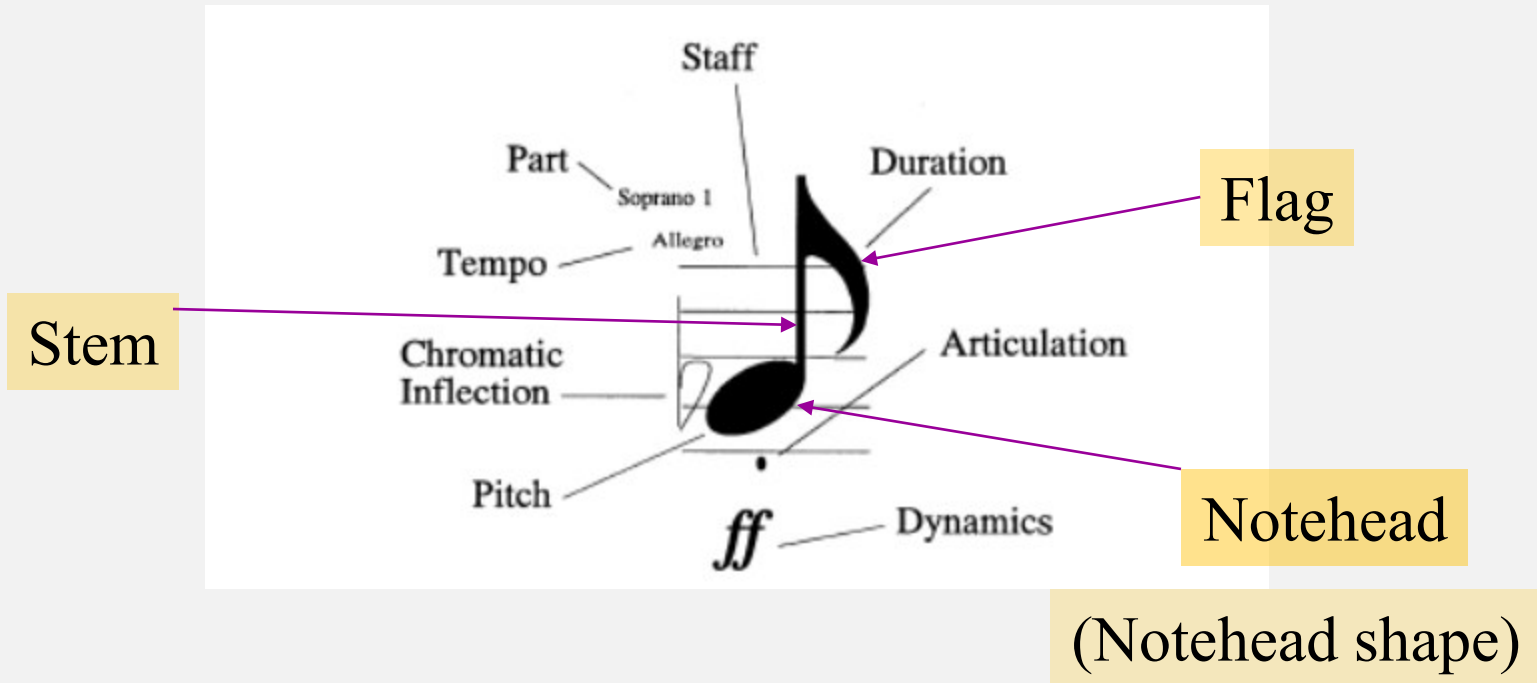
## DURATION

### **Assembly of objects:**

- Noteheads
- Stems and flags
- Beams
- Slurs

**Value inferred from  
combinations of objects**

# SCORE's approach to the note



*Essential Dictionary of Music Notation* (1996; recommended): See <http://www.alfred.com/Products/Essential-Dictionary-of-Music-Notation--00-16638.aspx>

# SCORE's approach to systems

Data-entry order

1. Encode lowest voice
2. Encode other voices on first system
3. Repeat until all systems are encoded
4. Assemble page(s)

Start here

The image shows a musical score for a string quartet. The instruments are listed on the left: clarinet in A, violino I, violino II, viola, and violoncello. The violoncello part is highlighted with a purple background. To the right of the score is a vertical green box containing the numbers 5, 4, 3, 2, 1, indicating the data-entry order from top to bottom. Below the first system, there is a double bar line and a second system of music.

5  
4  
3  
2  
1

# Part/score orientation in SCORE

## Process

1. Encode lowest voice
2. Encode other voices on first system
3. Repeat until all systems are encoded
4. Assemble page

**Implications:**  
**Content must be known in advance**

From whom is SCORE intended?

The image displays a musical score snippet with five staves. The staves are labeled on the left as 'clarinet in A', 'violin I', 'violin II', 'viola', and 'violoncello'. The 'violoncello' staff is highlighted with a purple background. The score is written in 2/4 time and features various musical notations, including notes, rests, and dynamic markings like 'f' and 'p'. A vertical bar line is present in the middle of the system. The bottom part of the image shows a continuation of the score with a double bar line and a section marked '8'.

# SCORE input/editing system

ASCII (computer keyboard) input (next slide)

Parametric editing (2)

Two stages:

- Data entry
- Data editing

The screenshot shows the WinScore software interface. At the top, there is a menu bar with 'File', 'View', 'Format', and 'Help'. Below the menu bar is a table with columns for 'Item #', 'P1/11', 'P2/12', 'P3/13', 'P4/14', 'P5/15', 'P6/16', 'P7/17', 'P8/18', 'P9/19', and 'P10/20'. The table contains the following data:

Item #	P1/11	P2/12	P3/13	P4/14	P5/15	P6/16	P7/17	P8/18	P9/19	P10/20	
189/268	NOTE	1	5.0	60.05	13.00	10.00	0.000	0.250	3.500	0.000	0.000
			-5.250								

Below the table is a musical score for a piano piece. The score is written on a grand staff with treble and bass clefs. It features a key signature of one sharp (F#) and a time signature of 9/8. The score includes dynamic markings such as *ff*, *mp*, and *f*. There are also performance instructions like '13' and '8' above the staff, and '5:3' and '5:4' below the staff. The score is displayed on a white background with black notation.



# SCORE input order

---

## Pitch

Height on a staff

Pitch (names)

## Pitch inflection

Marks (#, B, etc)

Rhythm (names)

## Pitch articulation

Marks (staccato, tr)

Marks

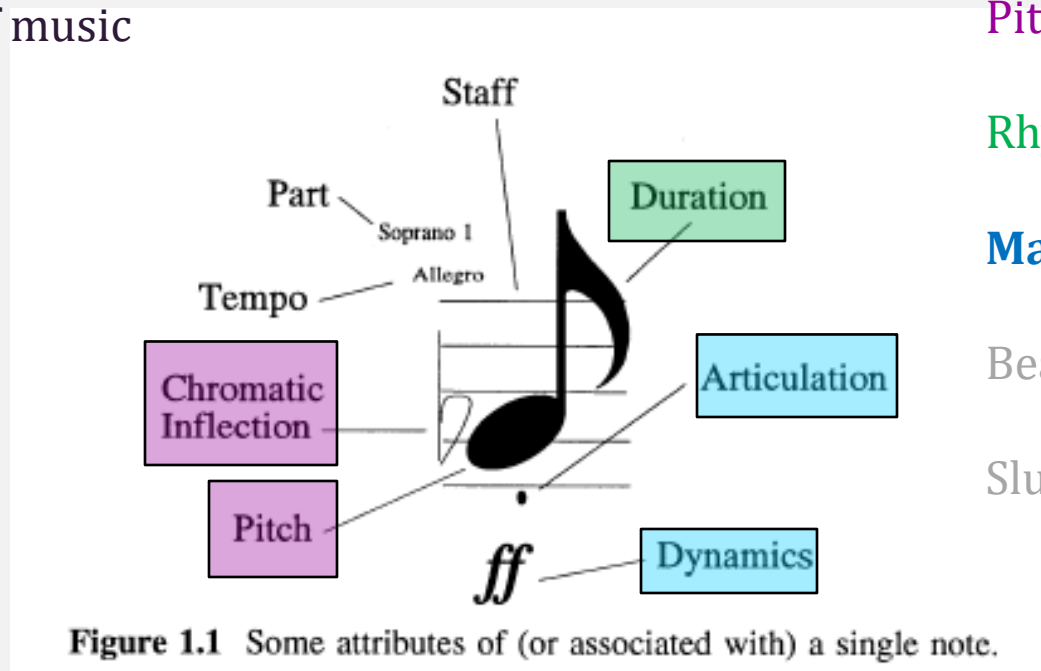
## Duration

Beams (positions)

Slurs (positions)

# Musical features of one note (SCORE)

A molecule of music



Pitch (names)

Rhythm (names)

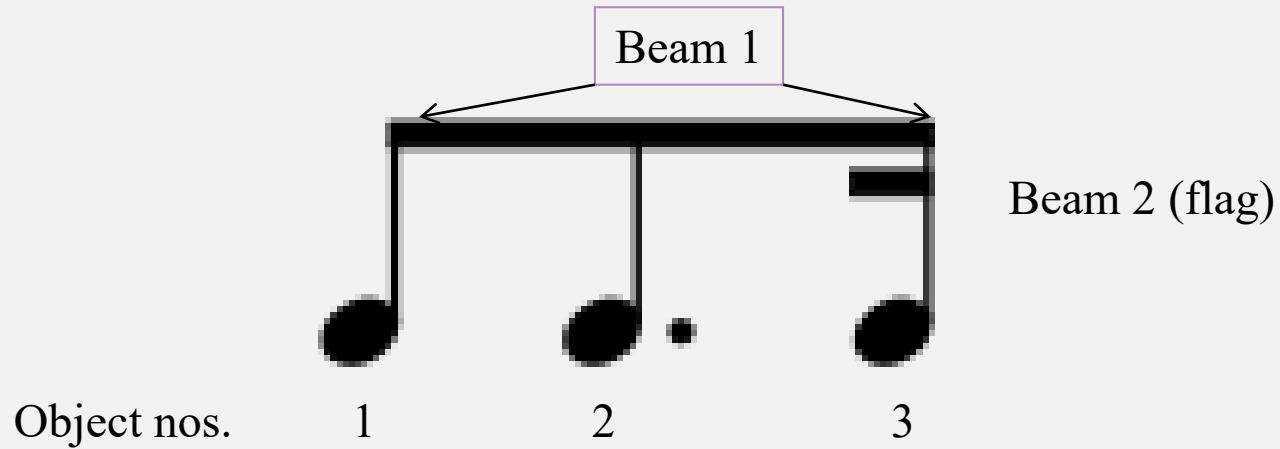
Marks

Beams (positions)

Slurs (positions)

# Object groups: (#4) beams

---



# Object groups: (#4) slurs

---

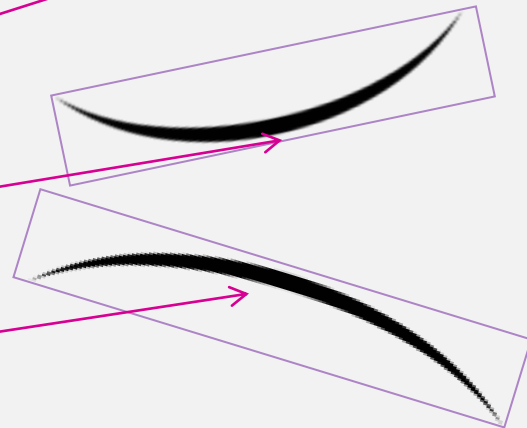
**Beams** vary by

- Length
- Inclination



**Slurs** vary by

- Length
- Inclination
- Inversion
- Apex (nadir) of arc  
(in relation to midpoint)

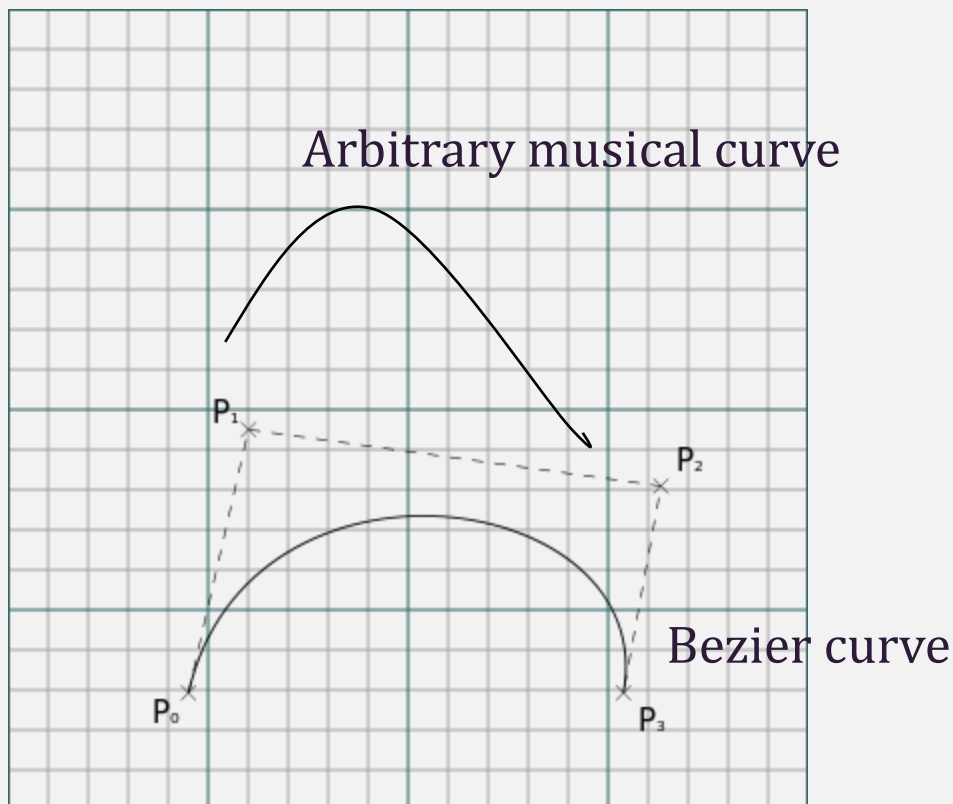


# SCORE's imaginary grid

---

**Slurs** vary by

- Length
- Inclination
- Inversion
- **Apex (nadir) of arc**  
(in relation to midpoint)



# Symbolic codes: DARMS and SCORE

## □ DARMS (1965)

- Columbia/Princeton/Yale
- **Theoretically** for notation
- One-pass **input** scheme
- Required **intermediate representation** for score assembly
- *Note Processor* (c.1986-92): only **commercial** program to use it [on PCs]
- **Legacy** = analysis

## □ SCORE (1972)

- **Stanford/Bell Labs**
- **Actually** for notation
- Five-pass **input** scheme
- Required **intermediate representation** for score assembly
- **SCORE** (1972--): only **commercial** notation program to use it
- **Legacy** = collected works of major composers

# Music V → SCORE

MUSIC V

SCORE

**Max Mathews**

Bell Labs (NJ)

*Sound-list generation*

**Leland Smith**

Stanford (CA)

*Note-list generation*



```
GEN2 3 0.0 2 1 1 0 .5 0 .25 0
NOT 1 2 0.0 .5 1 440
NOT 1 2 0.5 .5 2 554
NOT 1 2 1.0 .5 3 660
NOT 1 2 1.5 .5 4 554
NOT 1 2 2.0 1.0 5 880
NOT 1 2 3.0 .5 3 660
NOT 1 2 3.5 .5 2 554
NOT 1 2 4.0 .5 2 494
NOT 1 2 4.5 .5 4 588
NOT 1 2 5.0 1.0 5 740
```

This statement defines an instrument. Here a wave-form table for the clarinet is set up. GEN2 calls an oscillator subroutine. The parameters are (1) an operation code [3 = generate function], (2) an action time, (3) an instrument number, (4) a table number, and (5-10) the relative amplitudes of harmonics 1..6.

These statements cause notes to be played. The initial parameters are (1) an operation code [1 = play note], (2) an instrument number, (3) a start-of-action time, (4) event duration, (5) an absolute amplitude for the event, and (6) event frequency (Hz). These parameters may be followed by a variable number of user-defined parameters (not shown).

**Example G1** Music V representation of Bars 1 and 2 of the *Clarinet* part of the Mozart trio.

# SCORE/CCRMA: A Brief History



DC Power Lab site

## Who? **Leland Smith (1925-2013)**

- Where? Stanford/CCMRA/Palo Alto
- Goal? engraving-quality music
- Method? ASCII input, screen editing

## Who uses SCORE?

- Schott (DE)
- CF Peters (DE)
- Ricordi (IT)
- Hal Leonard (US)
- Composers in Stanford sphere

CCRMA pre-history c. 1974

John Chowning

Leland Smith



Photo: Patti Wood



# Score vs MuseScore

---

## SCORE

**Pitch:** reliable

**Duration:** reliable

**MIDI playback:** [pre-MIDI]

**File import:** little

**Export:** limited

**System, score assembly:** manual

**Extensible symbol set:** extensive

**Fonts:** beautiful; non-Roman available

## MuseScore

**Pitch:** not always reliable

**Duration:** not always reliable

**MIDI playback:** yes

**File import:** MIDI, MusicXML, SCORE

**Export:** MusicXML

**Score assembly:** automatic\*

**Extensible symbol set:** limited

**Fonts:** recently improved by “Leland”

# Symbolic vs. MIDI-based notation

---

## SCORE-type program

**Pitch:** reliable

**Duration:** reliable

**MIDI playback:** [partial]

**File import:** little

**Export:** limited

**System, score assembly:** manual

**Extensible symbol set:** yes

## Finale-type program

**Pitch:** not always reliable

**Duration:** not always reliable

**MIDI playback:** yes

**File import:** MIDI, MusicXML, SCORE

**Export:** MusicXML

**Score assembly:** automatic

**Extensible symbol set:** partial