An Introduction to **Symbolic Music Processing** in Python with partitura

Carlos Cancino-Chacón, Emmanouil Karystinaios, Silvan David Peter, Francesco Foscarin

JOHANNES KEPLER UNIVERSITY LINZ

Institute of Computational Perception

Tutorial Organization

- Introduction to symbolic music processing

1. An introduction to the Partitura library
2. Automatic alignment between performances and scores.
3. Pitch spelling with Partitura.
4. Transformer Based Beat Generator using Partitura.

Theoretical presentation

Hands-on tutorial

Symbolic Music

Symbolically encoded Music

(aka symbolic music)

The subset of **musical data types** that **explicitly represent**:

- Note pitches
- Note onsets and durations
- Voices
- Key signatures
- …

Symbolically encoded Music

(aka symbolic music)

The subset of **musical data types** that **explicitly represent**:

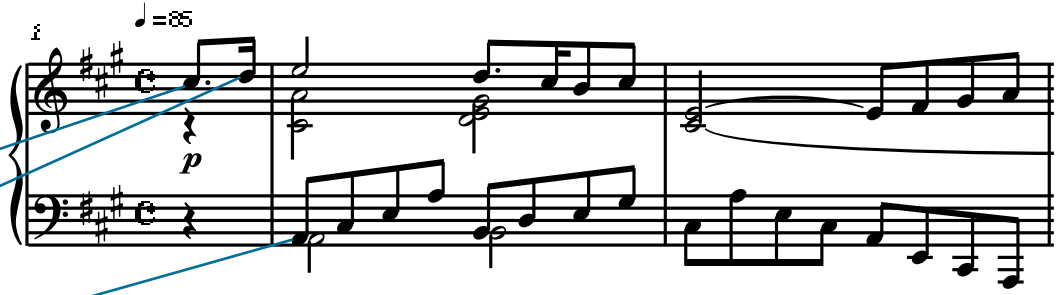
- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- …

Audio

Raster Image

Musical Score

- Note pitches (pitch spelling)
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- …



C#

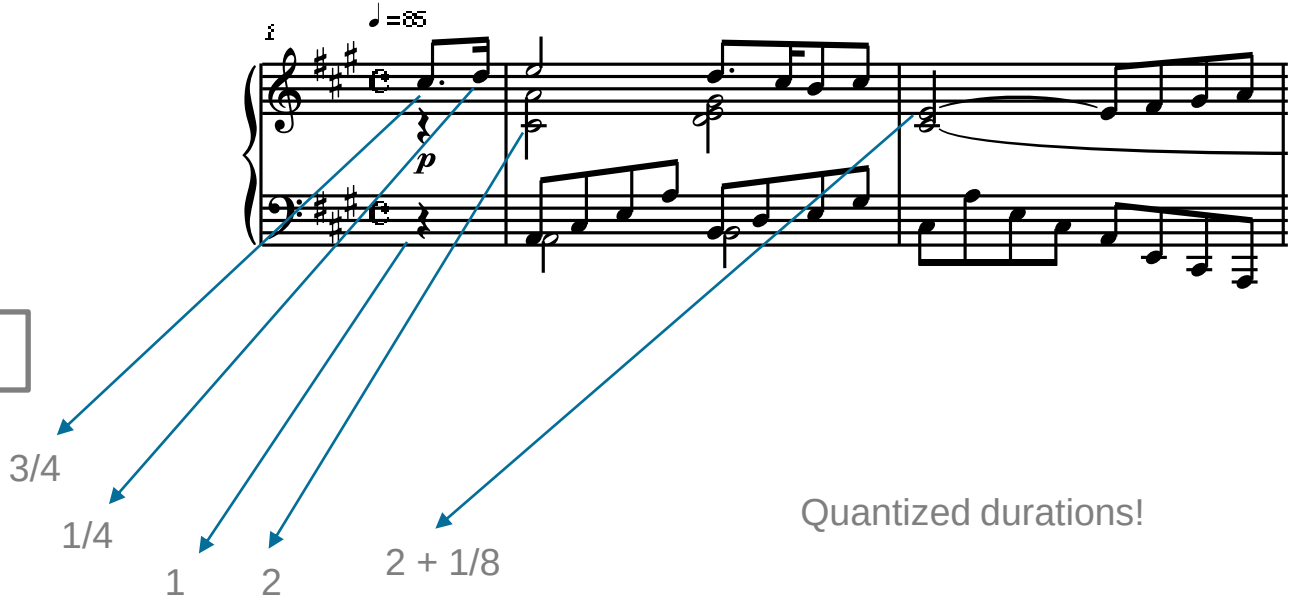
D

A

Musical Score

- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- ...

Musical units, e.g. 1 = ♩



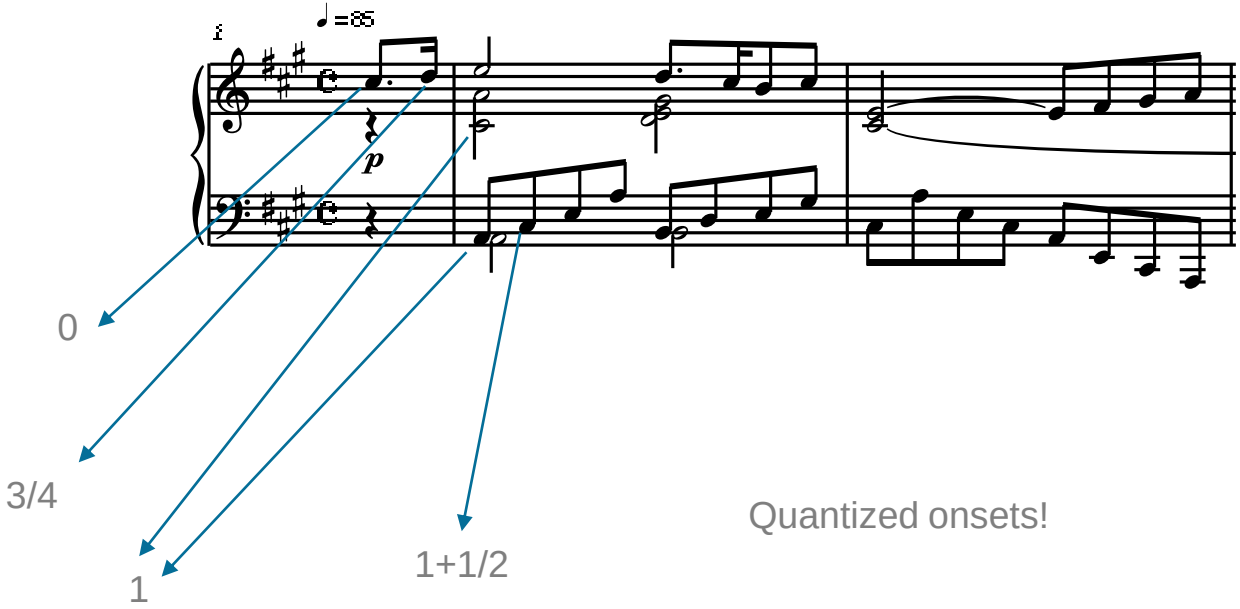
Quantized durations!

Musical Score

- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- ...

Musical units, e.g. 1 = ♩

Cumulative duration of previous events



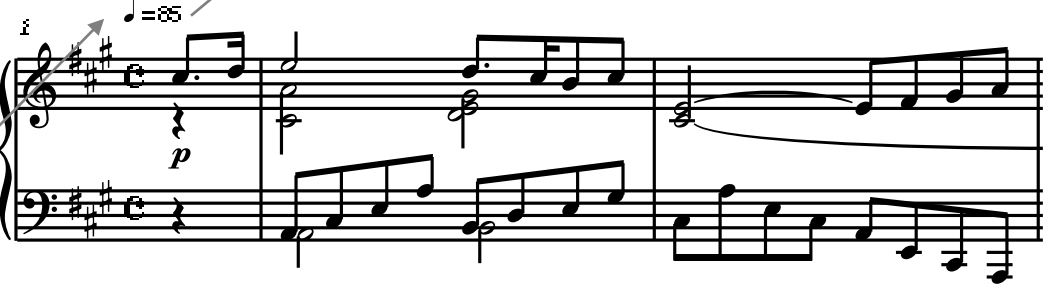
Quantized onsets!

Musical Score

- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- ...

Musical units, e.g. 1 = ♩

+



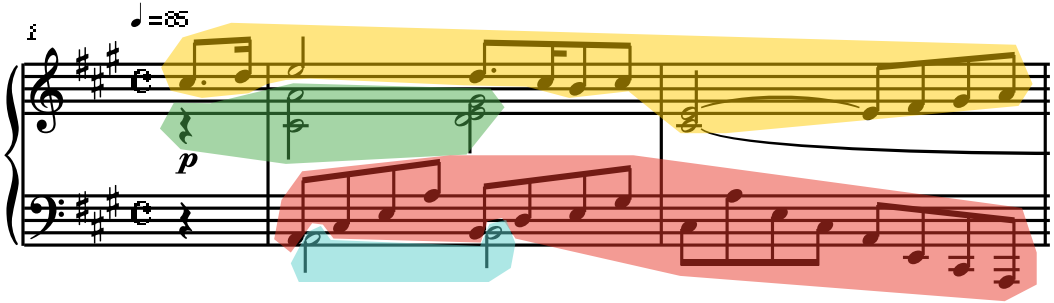
=

Real physical time, e.g. seconds

Quantized temporal positions!

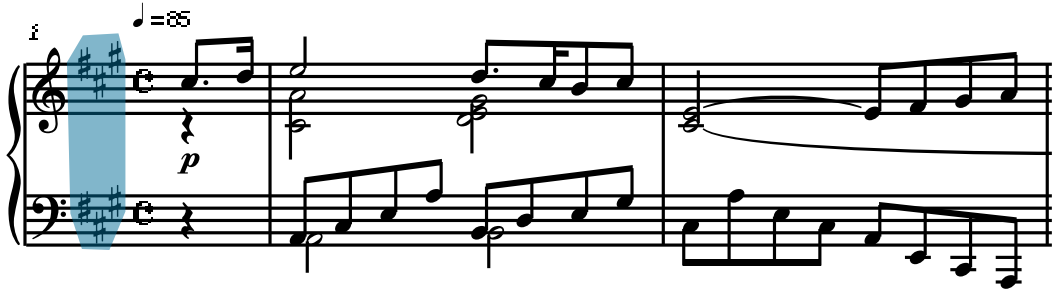
Musical Score

- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- …



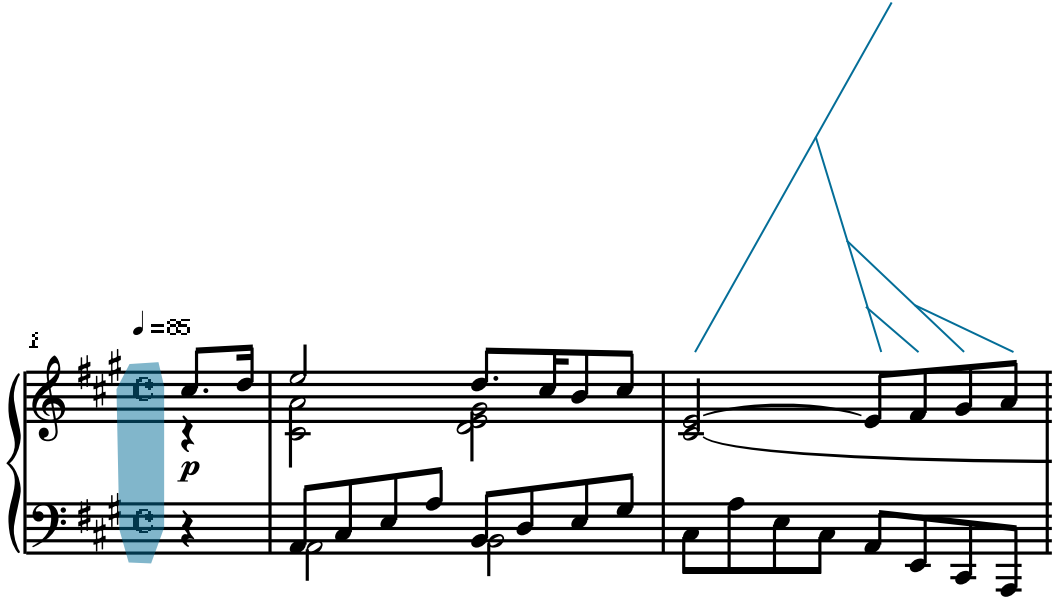
Musical Score

- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- …



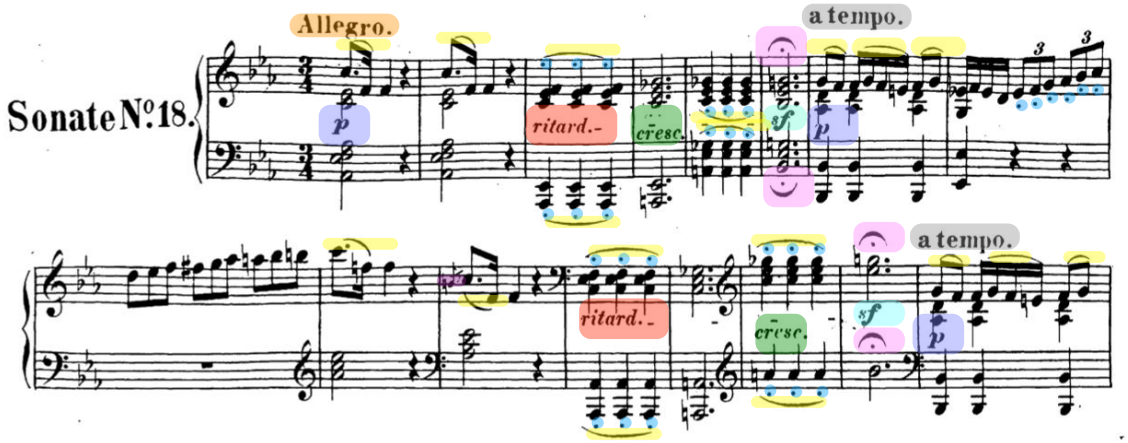
Musical Score

- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- …



Musical Score

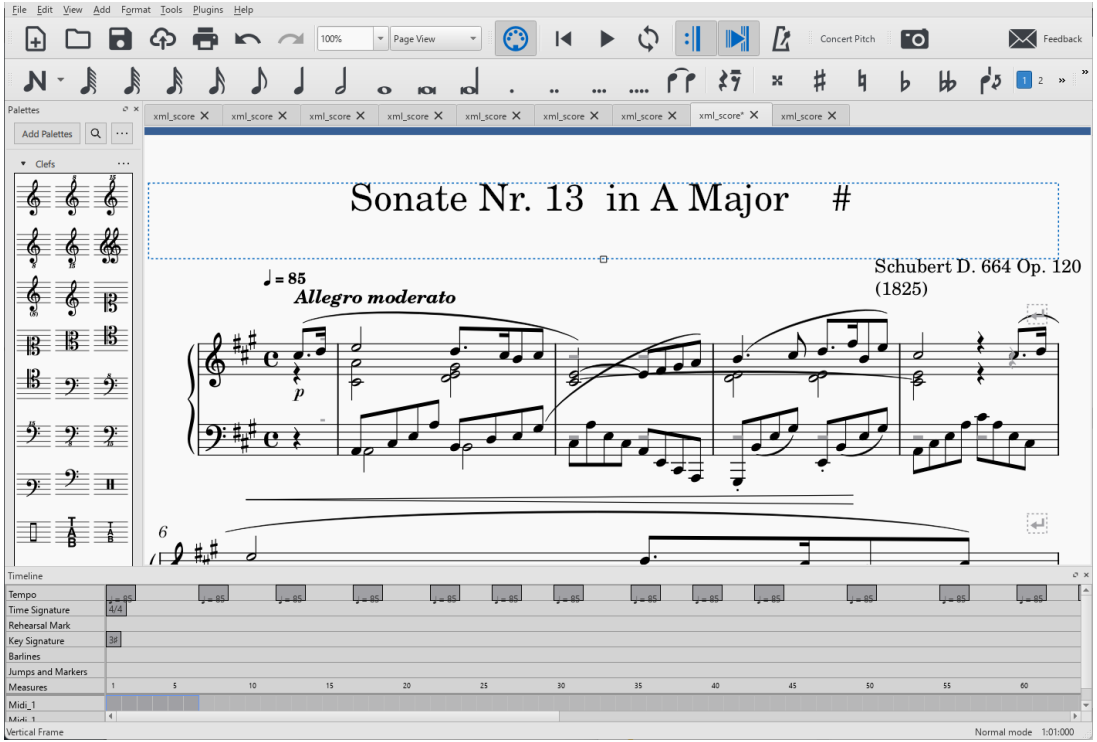
- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- …



Typically produced with a music notation software

Musical Score

- Note pitches
- Durations
- Onsets
- Voices
- Key signatures
- Metrical positions
- …

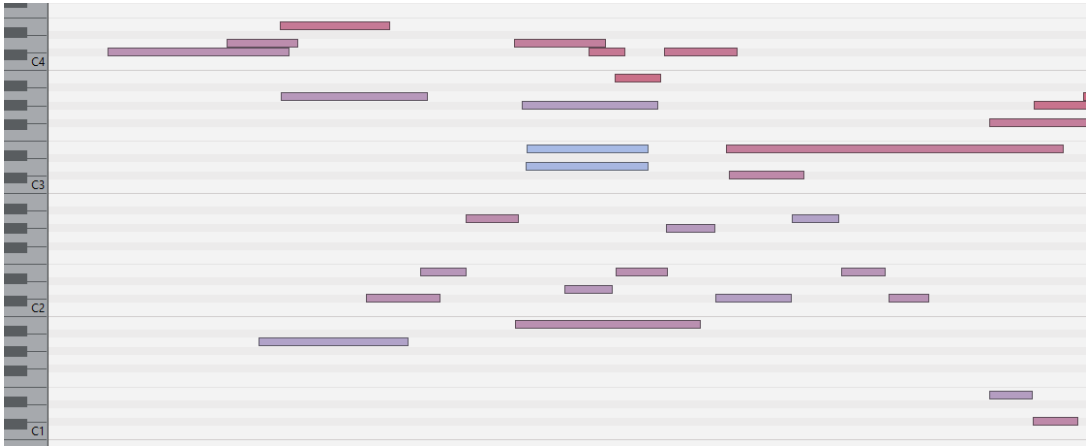


Musical performance

Typically produced performing an instrument

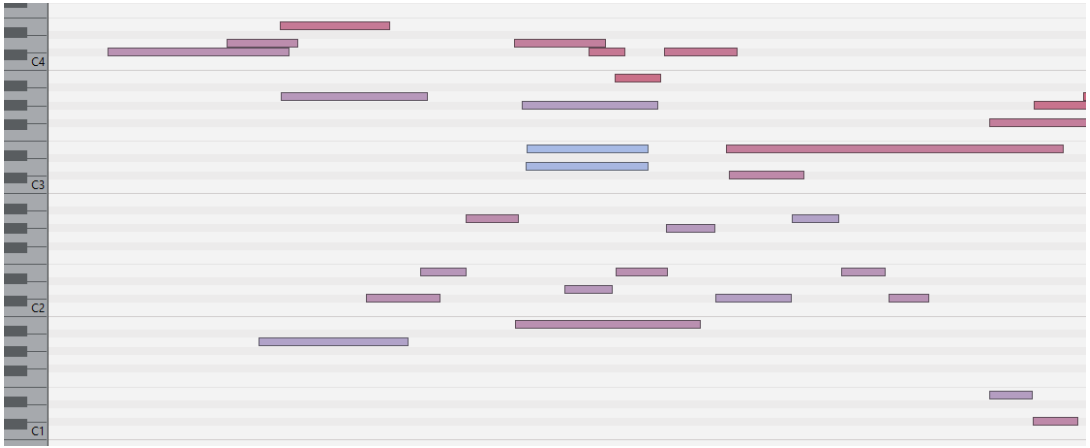
Musical performance (symbolic)

Instruments with MIDI sensors



Musical performance

- Note pitches (no pitch spelling)
- Onsets
- Durations
- Velocity



Musical performance

- Note pitches (no pitch spelling)
- Onsets
- Durations
- Velocity

Which key is pressed?

Musical performance

- Note pitches (no pitch spelling)
- Onsets
- Durations
- Velocity

Real physical time, e.g. seconds

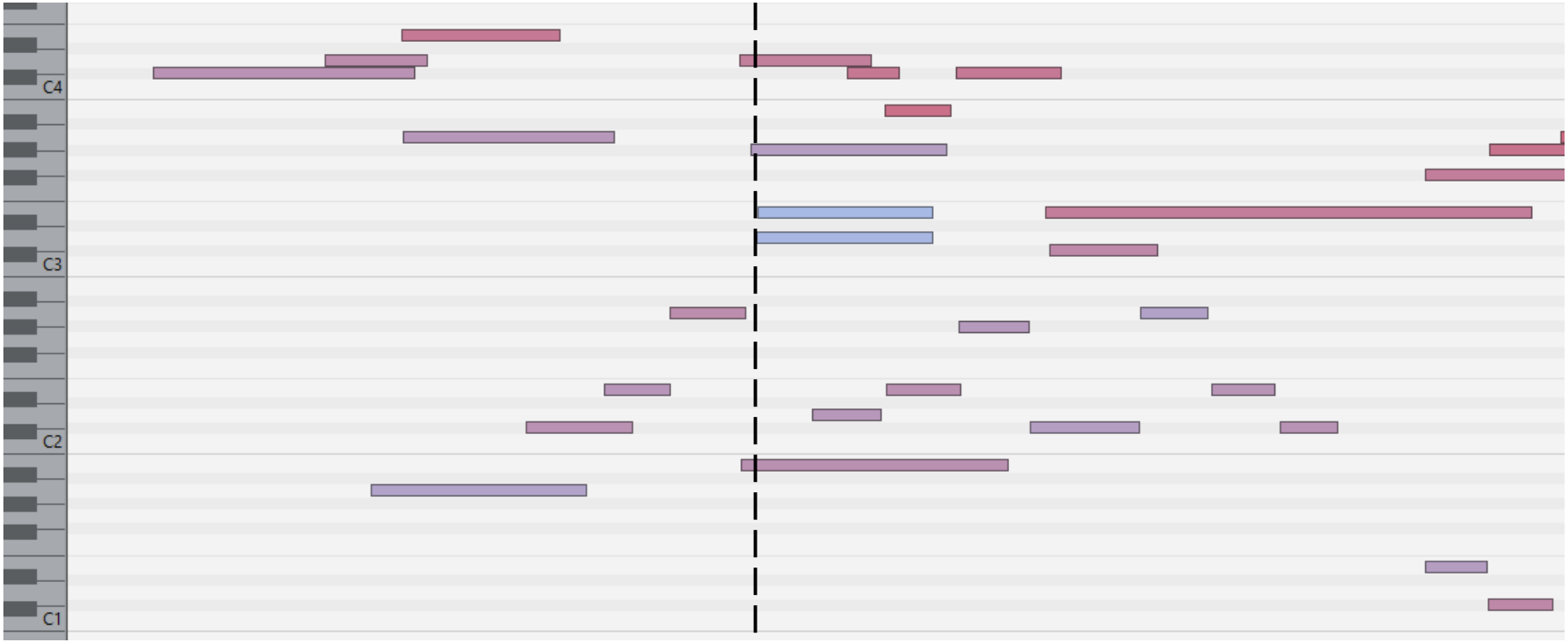
Unquantized temporal positions!

Musical performance

- Note pitches (no pitch spelling)
- Onsets
- Durations
- Velocity

Real physical time, e.g. seconds

Unquantized temporal positions!



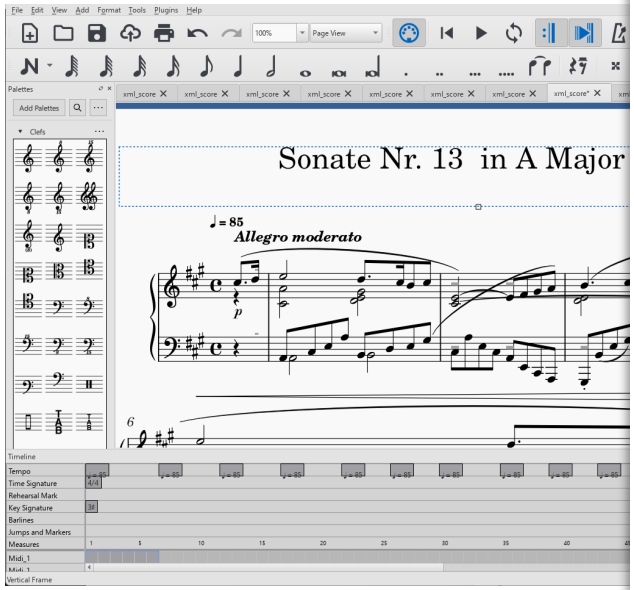
Unquantized temporal positions!

Musical performance

- Note pitches (no pitch spelling)
- Onsets
- Durations
- Velocity

How fast is the key pressed

Encoding Formats - musical score




```
</clef>
</attributes>
<note>
<pitch>
<step>E</step>
<octave>4</octa
</pitch>
<duration>1</dura
<voice>1</voice>
<type>quarter</ty
<stem>up</stem>
<staff>1</staff>
</note>
<note>
<pitch>
<step>D</step>
<octave>4</octa
</pitch>
<duration>1</dura
<voice>1</voice>
<type>quarter</ty
<stem>up</stem>
<staff>1</staff>
</note>
<note>
<pitch>
<step>C</step>
<octave>4</octa
</pitch>
<duration>1</duration>
<voice>1</voice>
```

```
<staffGrp xml:id="s1uvhvv9
<staffGrp xml:id="P1" b
<instrDef xml:id="i1
<staffDef xml:id="s1
<clef xml:id="c10
<keySig xml:id="k
<meterSig xml:id=
</staffDef>
<staffDef xml:id="s1
<clef xml:id="cql
<keySig xml:id="k12e1gun" mode="major" sig="3s" />
<meterSig xml:id="m1589vdf" count="4" sym="common" unit="4" />
</staffDef>
<grpSym xml:id="gxqo0lp" symbol="brace" />
</staffGrp>
</staffGrp>
</scoreDef>
<section xml:id="s14cxhpk">
<pb xml:id="p184jxzx" />
<measure xml:id="m1n3jher" n="1">
<staff xml:id="slsk790" n="1">
<layer xml:id="l14u14il" n="1">
<beam xml:id="b1r7ub6h">
<note xml:id="n5m2zh1" dots="1" dur.ppq="9" dur="8" oct="5" pname="c" stem.dir="up"
<note xml:id="nqyz76c" dur.ppq="3" dur="16" oct="5" pname="d" stem.dir="up" />
</beam>
</layer>
<layer xml:id="l5i0pcp" n="2">
<rest xml:id="rv1jq09" dur.ppq="12" dur="4" />
</layer>
</staff>
<staff xml:id="srdjrxw" n="2">
```

38	16En	16aaL	.	.
39	16G#JJ	16gg#JJ	.	.
40	16D#LL	16ff##LL	.	.
41	16G#	16gg#	.	.
42	16BB#	16ff#	[8dd#	.
43	16G#JJ	16gg#JJ	.	.
44	16C#LL	16ee#LL	8dd#L]	f
45	16D#	16gg#	.	.
46	16E	16een	8cc#	.
47	16F#JJ	16gg#JJ	.	.
48	16G#LL	16dd#LL	8b#	.
49	16A#	16gg#	.	.
50	16Bn	16ddn	8bnJ	.
51	16E#JJ	16gg#JJ	.	.
52	=3	=3	=3	=3
53	16F#LL	16cc#LL	8a#L	.
54	16G#	16gg#	.	.
55	16An	16b#	8anJ	.
56	16D#JJ	16ff#JJ	.	.
57	16EnLL	16b#LL	4g#	.

Encoding Formats – symbolic performance

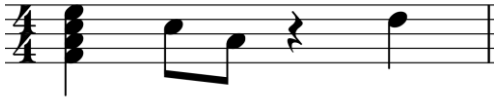
Midi

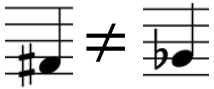
Delta-Time (decimal)	Event-Code (hex)	Other Bytes (decimal)
0	FF 58	04 04 02 24 08
0	FF 51	03 500000
0	C0	5
0	C1	46
0	C2	70
0	92	48 96
0	92	60 96
96	91	67 64
96	90	76 32
192	82	48 64
0	82	60 64
0	81	67 64
0	80	76 64
0	FF 2F	00

Scores vs. Performances

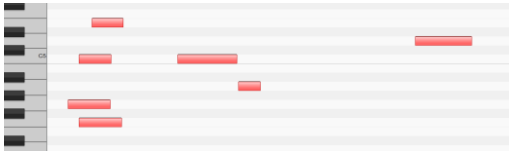
(in symbolic music)

Musical scores



- “Musical” time units, quantized
- Hierarchical time representation
- Pitch spelling 
- High-level musical information
  - Key/time signature
  - Voices
  - …

Musical performance



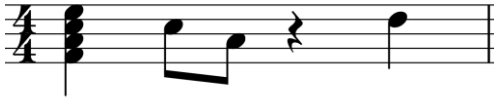
- Time expressed in seconds
- Sequential representation
- MIDI notes
- Precise timing and velocity of each note

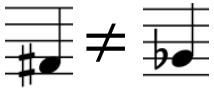
JOHANNES KEPLER UNIVERSITY LINZ

Scores vs. Performances

(in symbolic music)

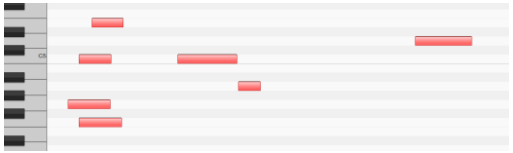
Musical scores



- “Musical” time units, quantized
- Hierarchical time representation
- Pitch spelling 
- High-level musical information
  - Key/time signature
  - Voices
  - …

File formats: musicxml, mei, kern

Musical performance



- Time expressed in seconds
- Sequential representation
- MIDI notes
- Precise timing and velocity of each note

File formats: midi

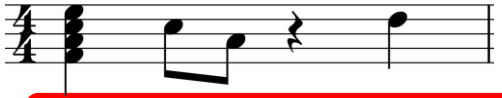
JOHANNES KEPLER UNIVERSITY LINZ

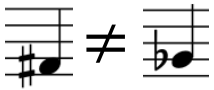
Scores vs. Performances

(in symbolic music)

MIDI Scores?

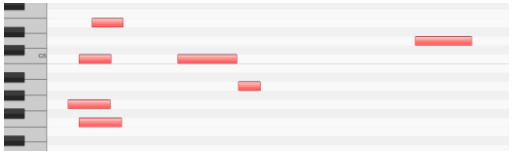
Musical scores



- "Musical" time units, quantized
- Hierarchical time representation
- ~~Pitch spelling~~ 
- High-level musical information
  - Key/time signature
  - Voices ?
  - ~~...~~

File formats: musicxml, mei, kern, (midi)

Musical performance



- Time expressed in seconds
- Sequential representation
- MIDI notes
- Precise timing and velocity of each note

File formats: midi

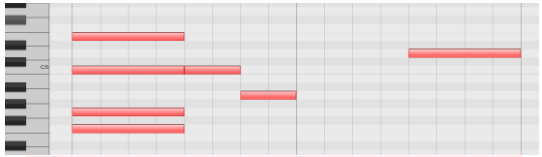
JOHANNES KEPLER UNIVERSITY LINZ

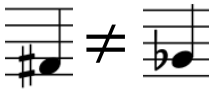
Scores vs. Performances

(in symbolic music)

MIDI Scores?

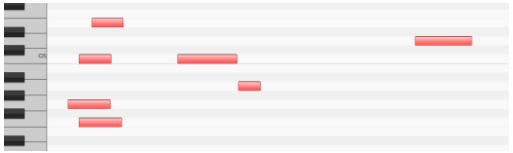
Musical scores



- "Musical" time units, quantized
- Hierarchical time representation
- ~~Pitch spelling~~ 
- High-level musical information
  - Key/time signature
  - Voices ?
  - ~~...~~

File formats: musicxml, mei, kern, (midi)

Musical performance



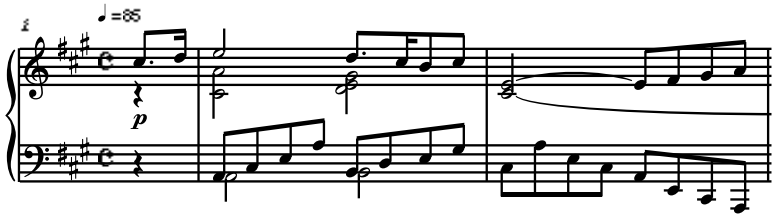
- Time expressed in seconds
- Sequential representation
- MIDI notes
- Precise timing and velocity of each note

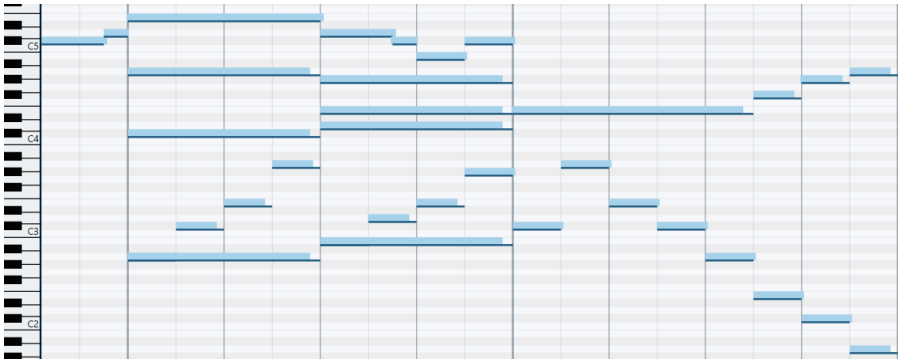
File formats: midi

JOHANNES KEPLER UNIVERSITY LINZ

Scores vs. Performances

MIDI Scores?

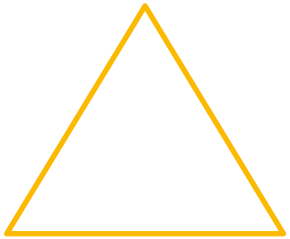




Scores vs. Performances

Babbitt's Representation
Domains of Music (1965)

Performance:
the produced musical experience



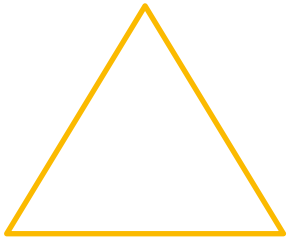
Score:
written musical experience

Auditory:
received musical experience

Scores vs. Performances

Babbitt's Representation
Domains of Music (1965)

Performance:
the produced musical experience

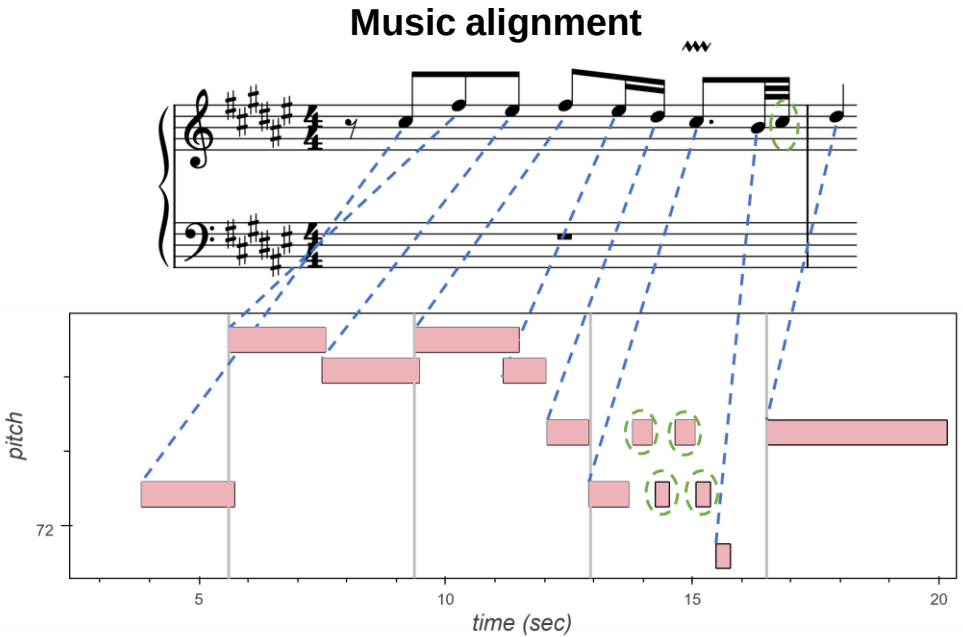


Score:
written musical experience

Auditory:
received musical experience

Score-to-performance alignments

Another symbolic data type



File formats:

Matchfile

Why using symbolic music

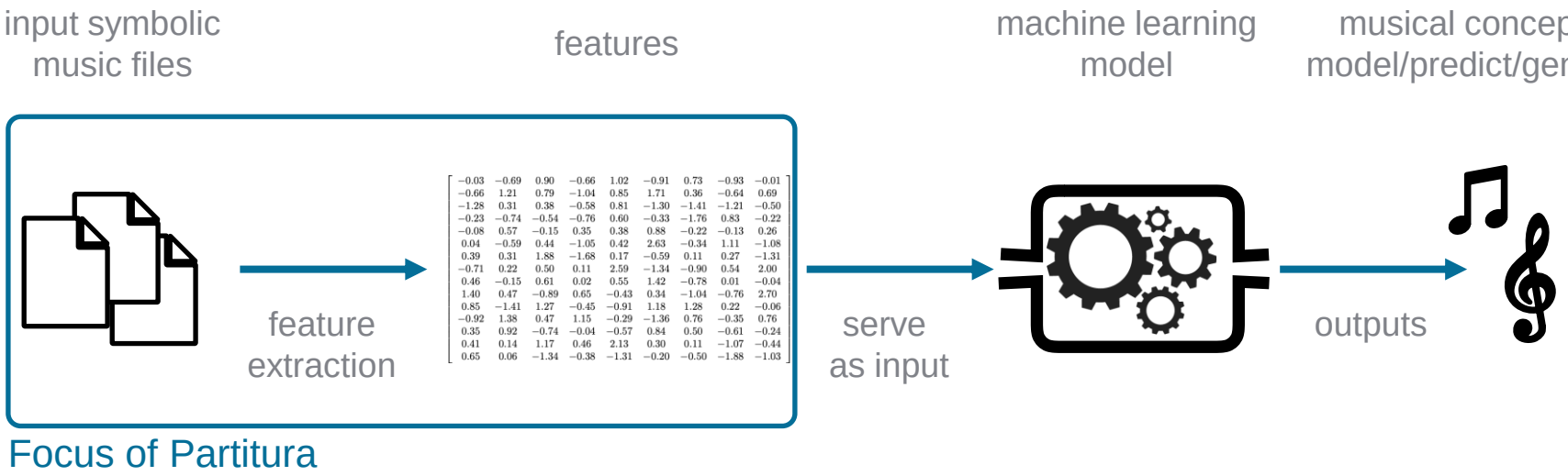
- “Less noisy” representation of musical content
- Better interpretation
- Can point to groups of explicitly encoded events (audio is “vertically mixed”)
- Usage as ground truth for high-level musical element prediction

Why scores over MIDI

- Higher level musical features available
- Readable by humans

MIR with symbolic music

A Common MIR Pipeline



Symbolic Music Processing Packages in Python

PrettyMIDI

- Focus on fast extraction of information from MIDI files
- No real distinction between performance and score
- Very easy to use for people without musical background

Raffel and Ellis

Partitura

- Simple, yet complete representation of scores
- Handling scores, performances and alignments
- Focus on lightweight extraction of MIR features

Music21

- Complete and hierarchical representation of scores
- Focus on computational musicology
- Very powerful but has a steep learning curve

Cuthbert & Ariza

Datasets

Name	Scores	Performances	Note-alignments
Vienna 4x22	4	88	43,450
Extended ASAP	222	1062	7,275,074

The Partitura Development Team

Current

Carlos Cancino-Chacón

Silvan Peter

Francesco Foscarin

Emmanouil Karystinaios

Patricia Hu

Tutorial presenters

Former

Thassilo Gadermaier

Nimrod Varga

Benevolent Dictator for Life

Maarten Grachten

https://cpjku.github.io/partitura_tutorial/