

Laurent Pugin

Swiss RISM / Fribourg University
laurent.pugin@rism.ch.org

Johannes Kepper

University of Paderborn
kepper@edirom.de

Perry Roland

University of Virginia
pdr4h@eservices.virginia.edu

Maja Hartwig

University of Paderborn
maja.hartwig@gmx.de

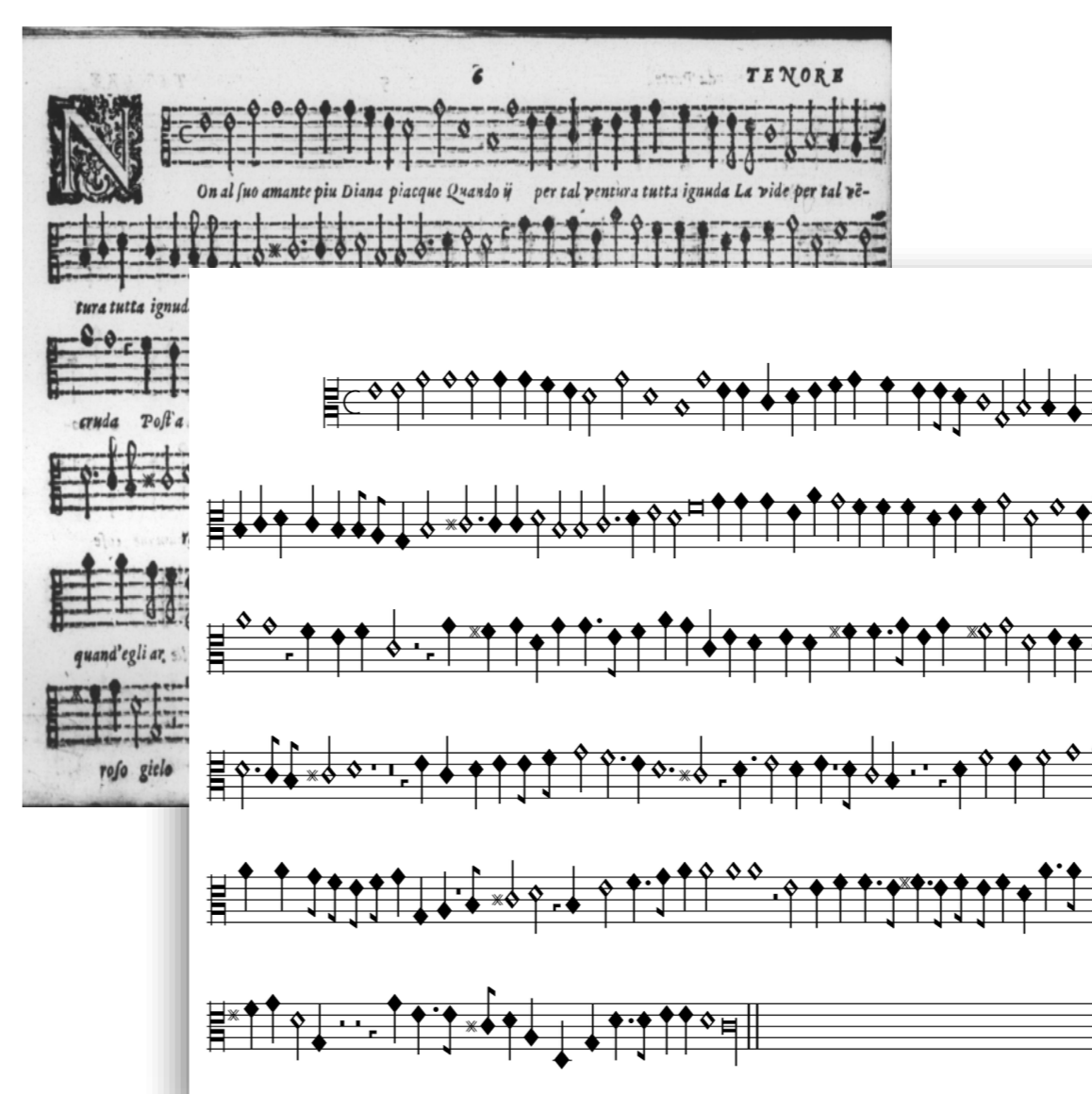
Andrew Hankinson

McGill University, Schulich School of Music
andrew.hankinson@mail.mcgill.ca

Introduction

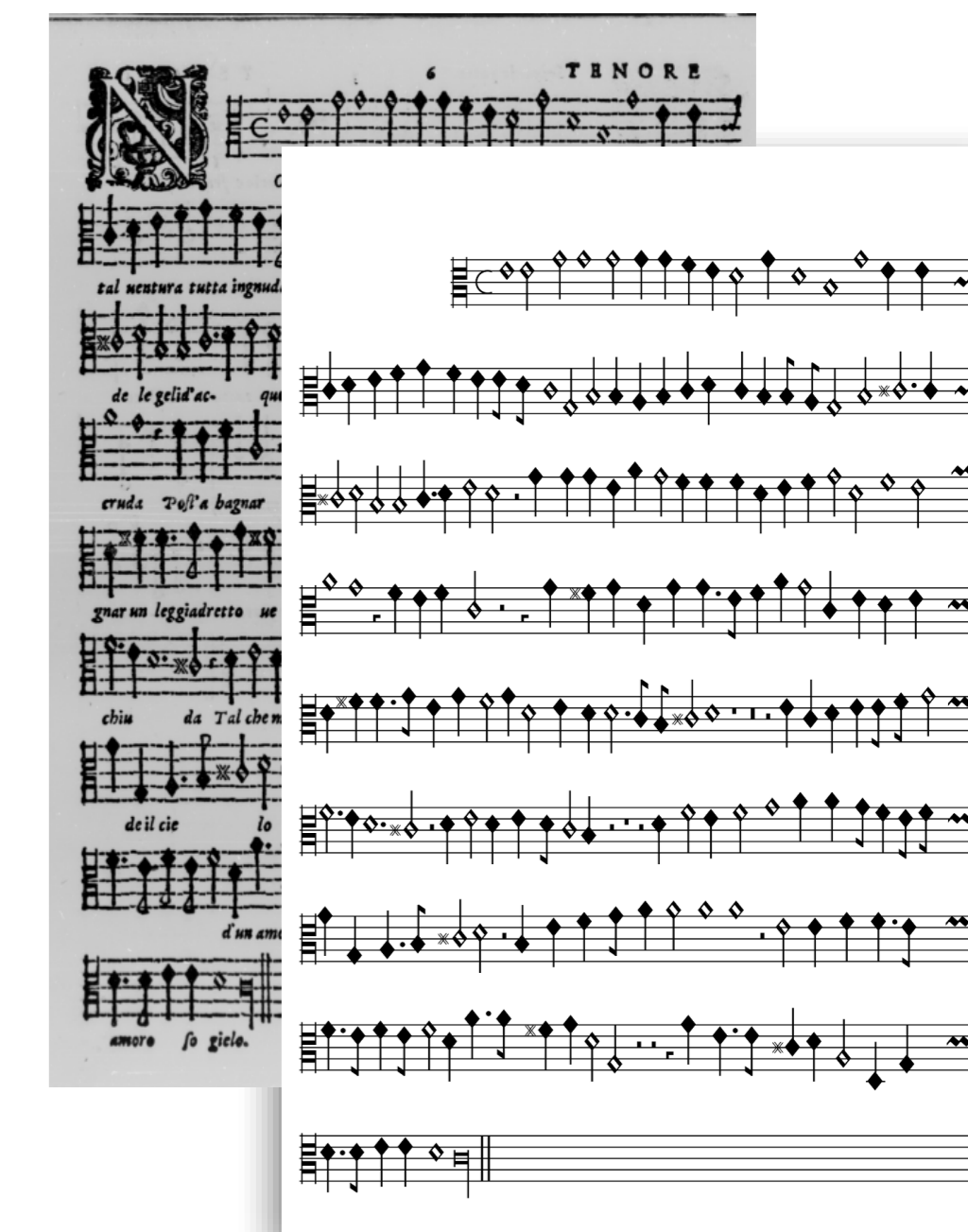
- Symbolic music notation can be separated into two components: Musical content—the “essence” of the music—and presentation—how symbols are arranged on a page or a computer screen.
- Some notation encoding systems do not separate these two components. This creates challenges for computational analysis of the musical content, since critical musical information is encoded solely in the visual domain. Similarly, this presents difficulties for capturing different visual appearances for the same musical content in, for example, different printed editions of the same piece of music.
- We describe a method in the Music Encoding Initiative format for separating musical content from presentation. This is especially useful when encoding multiple versions of the same musical content; for example, two editions of the same book, or a composer’s autograph and a printed edition.
- We achieve this by defining multiple presentation “sub-trees” separate from the encoded musical content. Each sub-tree defines a different layout for the same musical content, and the musical content remains free of any layout-specific information (like staff positions, note positions, system breaks, or page breaks).

Facsimile of source A

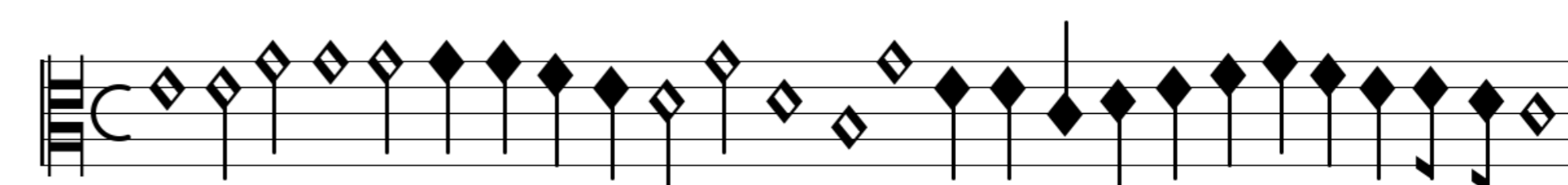


Layout of source A

Facsimile of source B

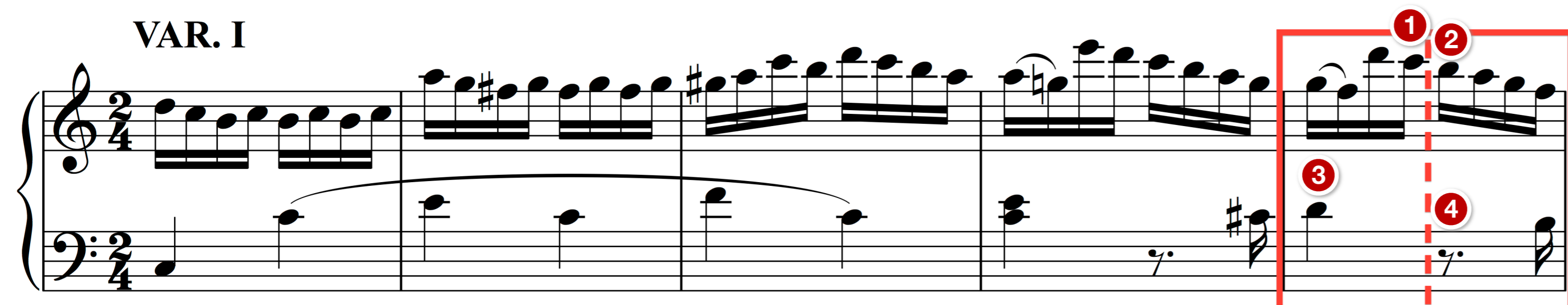


Layout of source B



Musical content of sources A and B

Figure 1: Source A (left) and source B (right) are different editions of the same piece of music. They have the same musical content but different layouts.



```
<measure n="5" xml:id="m5">
  <staff n="1" xml:id="s5s1">
    <layer n="1" xml:id="s5s1l1">
      <beam >
        <note xml:id="m5s1l1e1" pname="g" oct="5" dur="16" />
        <note xml:id="m5s1l1e2" pname="f" oct="5" dur="16" />
        <note xml:id="m5s1l1e3" pname="d" oct="6" dur="16" />
        1 <note xml:id="m5s1l1e4" pname="c" oct="6" dur="16" />
      </beam>
      <beam >
        2 <note xml:id="m5s1l1e5" pname="b" oct="5" dur="16" />
        <note xml:id="m5s1l1e6" pname="a" oct="5" dur="16" />
        <note xml:id="m5s1l1e7" pname="g" oct="5" dur="16" />
        <note xml:id="m5s1l1e8" pname="f" oct="5" dur="16" />
      </beam>
    </layer>
  </staff>
  <staff n="2" xml:id="m5s2">
    <layer n="1" xml:id="m5s2l1">
      <note xml:id="m5s2l1e1" pname="d" oct="4" dur="4" /> 3
      <rest xml:id="m5s2l1e2" dur="8" dots="1" /> 4
      <note xml:id="m5s2l1e3" pname="b" oct="3" dur="16" />
    </layer>
  </staff>
  <slur staff="1" startid="#m5s1l1e1" endid="#m5s1l1e2" />
</measure>
```

```
<page n="1">
  <system n="1">
    <laidOutStaff staff="1">
      <laidOutLayer>
        <!-- previous measures -->
        <!-- first half of measure 5 -->
        1 <laidOutElement target="m5s1l1e4" />
      </laidOutLayer>
    </laidOutStaff>
    <laidOutStaff staff="2">
      <laidOutLayer>
        <!-- previous measures -->
        <!-- first half of measure 5 -->
        3 <laidOutElement target="m5s2l1e1" />
      </laidOutLayer>
    </laidOutStaff>
  </system>
  <system n="2">
    <laidOutStaff staff="1">
      <laidOutLayer>
        <!-- second half of measure 5 -->
        2 <laidOutElement target="m5s1l1e5" />
        <!-- next measures -->
      </laidOutLayer>
    </laidOutStaff>
    <laidOutStaff staff="2">
      <laidOutLayer>
        <!-- second half of measure 5 -->
        4 <laidOutElement target="m5s2l1e2" />
        <!-- next measures -->
      </laidOutLayer>
    </laidOutStaff>
  </system>
</page>
```

Figure 2: Music notation and its representation in MEI. The numbers correspond to the musical content (left), the the layout (right) and the symbolic representation (above). The dotted red line represents a system break in one source.

Implementation

Organization

- The layout information is stored in a dedicated sub-tree represented by a <layouts> element within the <music> element.
- The <layouts> element can contain an arbitrary number of <layout> elements, each of them describing a different visualization of the same musical content.

Referencing System

- The musical content is referenced in the layout sub-tree using the <laidOutStaff>, <laidOutLayer>, and <laidOutElement> tags. Each of these provides a description of the positioning for the musical content.

Overlapping Hierarchies

- Elements in the layout sub-tree point “into” the musical sub-tree. This allows the layout tree to describe multiple independent layouts of the same musical content.

Content Selection

- To point to musical content, the <laidOutElement> tag features an @target attribute which contains the ID of the element whose position it is describing. See Figure 2 for an example.

Conclusion and Future Work

- The MEI Layout module is designed to separate content from presentation in musical markup.
- We have demonstrated how our module can be used to provide multiple independent layouts for the same musical content.
- By separating content and presentation, we make automated machine processing of MEI easier by ensuring musical information is encoded in the semantic domain, not the visual domain.
- The beta version of the layout module is developed within the MEI Incubator.
- The layout module is currently under review for inclusion in the core MEI distribution.

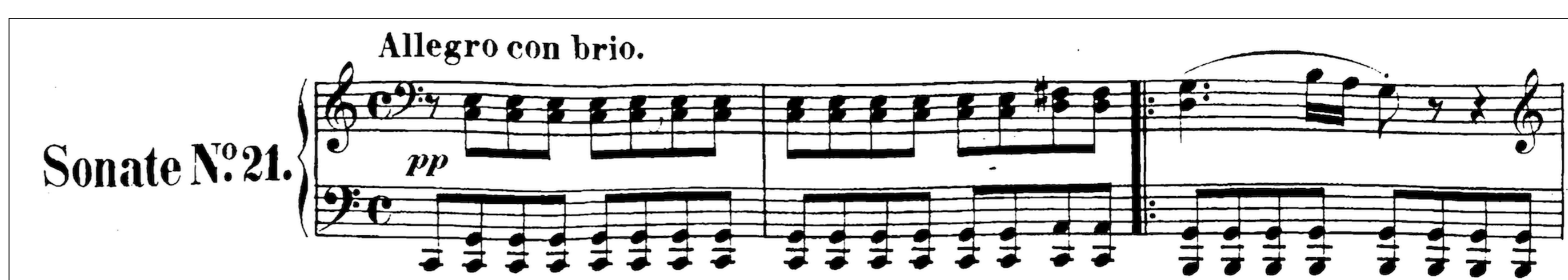
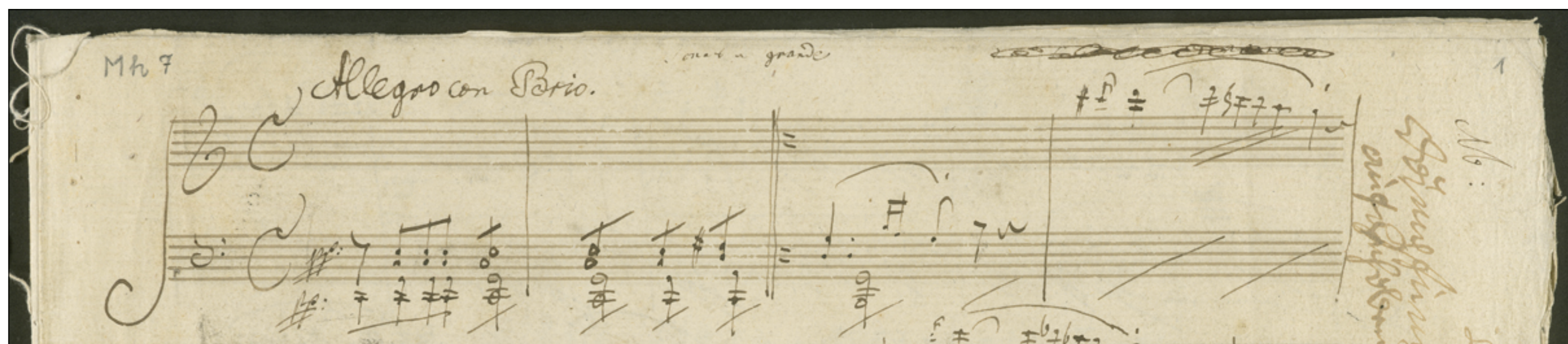


Figure 3: The beginning of Beethoven’s “Waldstein” sonata No. 21. The right hand is written on the lower staff in the manuscript and on the upper one in the edition.