MusicXML & the MEI

John Ashley Burgoyne

MUMT 611

18 January 2007

Music in XML

- Musical scores suggest gestures to musicians.
- MIDI is fundamentally a gestural interface.
- It is impossible to map performed gestures back onto musical directions precisely.
- We need other standards for scores.

Outline

- two standards using XML DTDs:
  - MusicXML
  - Music Encoding Initiative (MEI)
- common limitations

MusicXML

- developed by Michael Good (Recordare)
- strives to be a universal interchange format
- “designed to be sufficient, not optimal”
- replacement for Humdrum or MuseData?
- lacks a stand-alone viewer

MusicXML Example

```
<attributes>
  <divisions>4</divisions>
  <key>
    <fifths>-1</fifths>
    <mode>major</mode>
  </key>
  <time>
    <beats>2</beats>
    <beat-type>4</beat-type>
  </time>
  <staves>2</staves>
  <clef number="1">
    <sign>G</sign>
    <line>2</line>
  </clef>
  <clef number="2">
    <sign>F</sign>
    <line>4</line>
  </clef>
</attributes>
<note default-x="111">
  <pitch>
    <step>C</step>
```

Typical Musical Score

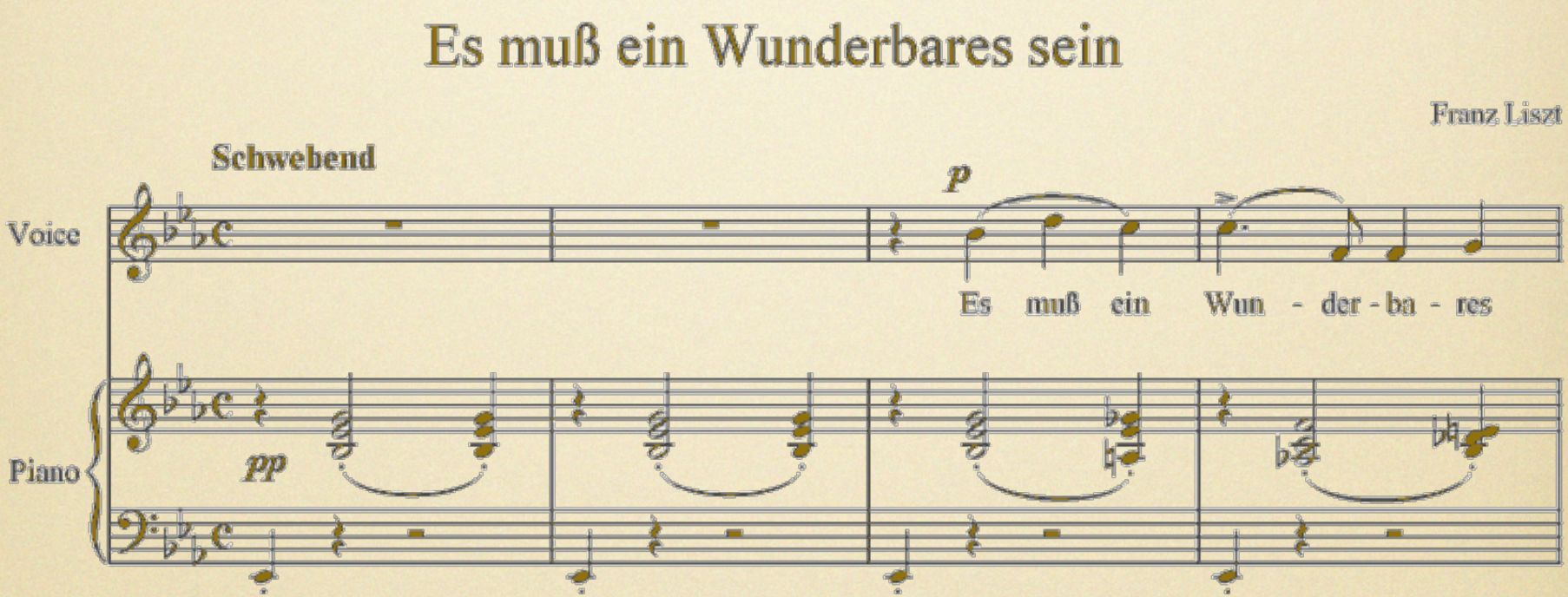

Figure 3: Original Music as Entered in Sibelius 4.1

Good 2006

Limitations of MIDI

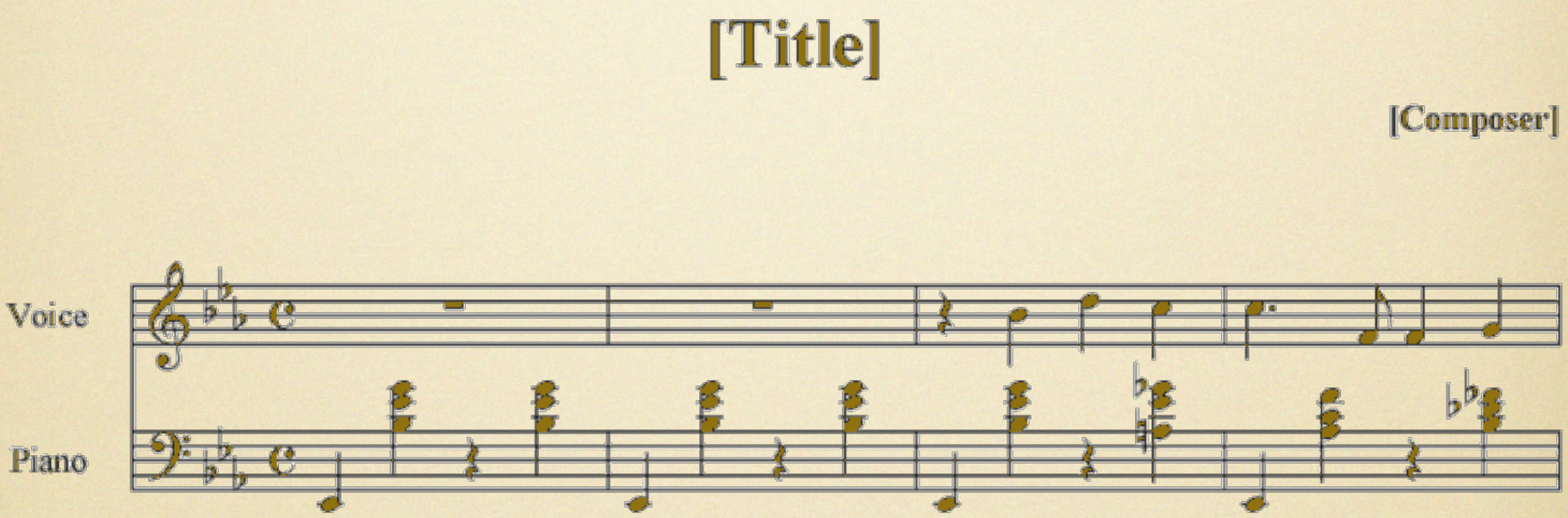

Figure 4: Music as Transferred to Finale 2007 via Standard MIDI File

Good 2006

MusicXML Transfer

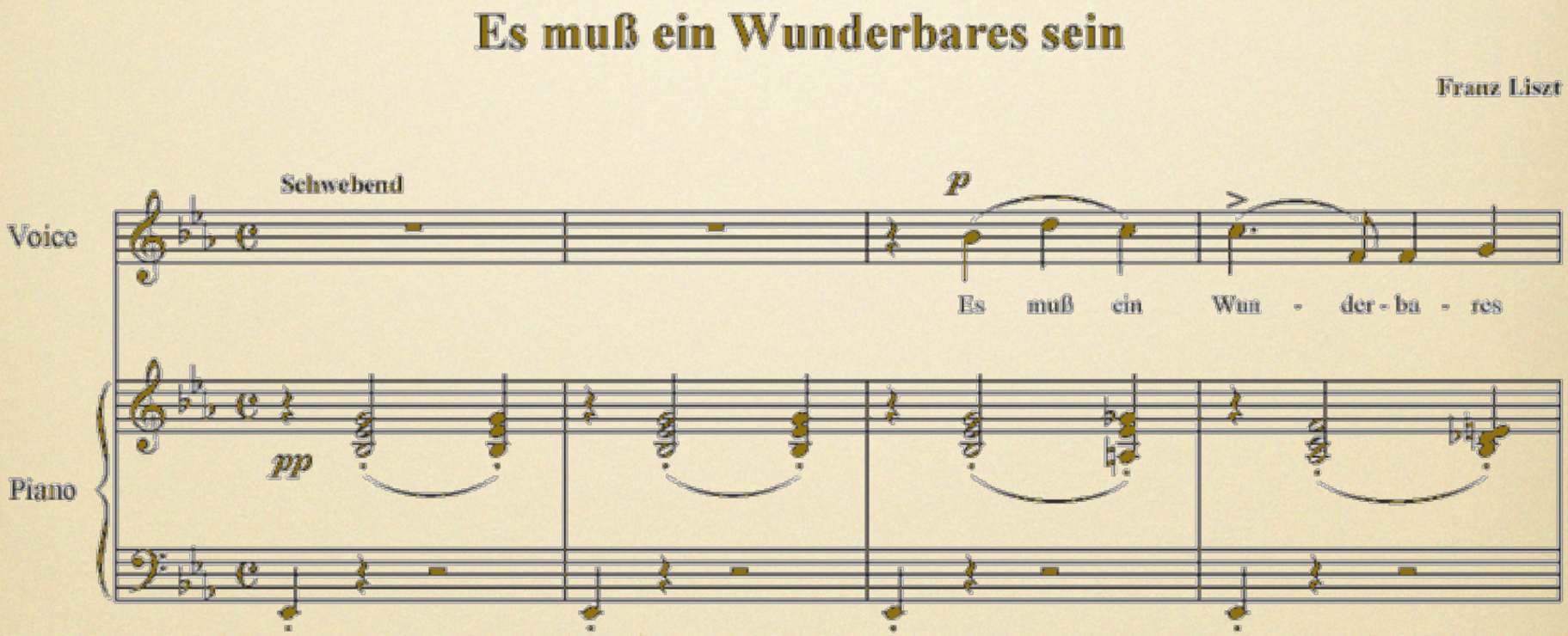

Figure 5: Music as Transferred to Finale 2007 via MusicXML

Good 2006

Commercial Success?

Good 2006

Lessons

- API usability
- bi-directional exchange
- overhead from standards organisations
- compare to NIFF and SDML

Music Encoding Initiative (MEI)

- developed by Perry Roland (UVA)
- inspired by the Text Encoding Initiative (TEI)
- similar in goals to MusicXML
- more idealised, less commercialised
- less commercial support, no direct viewer

Design Goals

- comprehensive
- declarative
- explicit
- interpreted
- hierarchical
- formal
- flexible
- extensible

MEI Example

```
<score>
   <scoredef>
      <pghead1><fw><fwr><fwd></fwd><fwd></fwd><fwd>9</fwd></fwr></fw></pghead1>
      <pgfoot1><fw><fwr><fwd>7545</fwd></fwr></fw></pgfoot1>
      <staffgrp symbol="brace" label.full="">
         <staffdef n="1" clef.line="2" clef.shape="G"/>
         <staffdef n="2" clef.line="4" clef.shape="F"/>
      </staffgrp>
   </scoredef>
   <section>
      <sectiondef key.sig="3s" meter.count="12" meter.unit="8"/>
      <measure n="1">
         <staff def="1">
            <layer def="1">
               <beam>
                  <note id="n1" pname="c" oct="6" dur="16" stem.dir="down"/>
                  <note pname="a" acci="s" oct="5" dur="16" stem.dir="down"/>
                  <note pname="f" oct="5" dur="16" stem.dir="down"/>
                  <note pname="g" oct="5" dur="16" stem.dir="down"/>
                  <note pname="b" acci="s" oct="5" dur="16" stem.dir="down"/>
                  <note pname="a" oct="5" dur="16" stem.dir="down"/>
               </beam>
```

TabXML

- project to encode lute tablature using XML
- could not use MusicXML or the MEI:
  - too hierarchical
  - focus on common music notation
- can be translated to MusicXML
- would like to interoperate with the MEI