

# SIMSSA DB: Go Jump in the (Data) Lake

**Cory McKay**, Marianopolis College, Canada

**Rebecca Mizrahi**, McGill University, Canada

# Topics

- Overview and goals of the SIMSSA DB
- Some highlights
  - Metadata
  - Auto-extracted, searchable features
  - Abstract works, sections and parts
  - Sources and provenance
- SIMSSA DB / LinkedMusic integration
- Live demo by Rebecca Mizrahi

# What is the SIMSSA DB?

- **Collaborative** database **prototype infrastructure** for holding and accessing **symbolic music files**, associated auto-extracted content-based **feature values**, and **musicologically-focused metadata**
  - With a web Django-based browser interface
- Populated by:
  - **Now:** Samples from research datasets we have constructed
  - **Medium-term:** Import existing open symbolic datasets that musicologists, libraries and others have already constructed
    - We can import such datasets, or users can **contribute them directly**
  - **Long-term:** Auto-population via (verified) OMR
- Focused (for now) on **early music**

# An infrastructure, not a corpus

- The SIMSSA DB is **not** intended just as a repository of music we have transcribed ourselves
  - Although are seeding it with datasets we have made, such as JLSDD (Cumming et al. 2018), Florence 164 (Cumming & McKay 2018), etc.
- Rather, it is a **general unified infrastructure** to which it is hoped **other scholars** can **contribute** and share symbolic music files (and more) that they have used in their own work

# SIMSSA DB prototype contribution form

## Create a Musical Work

### Title

Check if the work is already in the database. If so, then select it. If not, then check the "Musical Work not in database" checkbox below and enter the title in the field that appears. Please include opus number or catalogue numbers if applicable (e.g., Op. 55, D960, BWV 202).

Musical Work **not** in database

Title\*: [?](#)

Variant Titles: [?](#)

e.g. Eroica

Sections: [?](#)

1. Kyrie

## Genre(s)

What type of piece is this? (e.g., song, symphony, motet)

Type not in database

What style is this piece? (e.g., classical, jazz)

Style not in database

Sacred Or Secular:

## Medium of Performance

Please enter the instruments or voices below.

Instruments:

Instrument not in database

## Contributors [?](#)

Please complete one contributor before adding another. Who created the work? Use the drop-down menu to choose between different kinds of contributions. Add more contributors with the green button.

Contributor's Name:

Person is not in database

Role:

Certainty of attribution:

Certain

Uncertain

Unknown

Location:

Location not in database

e.g. Court of Marie V

Date of Contribution (range):

# Data quality

- Focus on **high-quality** data
- Quality of individual documents is especially important in **early music**:
  - Individual **details** very important to domain experts
    - e.g. a single cadence or even a single note
  - **Few extant sources**, so limited training/testing data will ever be available, and there is limited room for statistical noise
- **Problem**: Ensuring high-quality structured data requires expertise and effort on the part of contributors and validators
  - One of the reasons the SIMSSA DB is designed primarily for use by musicologists and, to a lesser extent, MIR researchers
- This tension between **quantity vs. quality** is not yet fully resolved; we may choose to find a different balance between them in the future
  - In terms of both the **amount of data** and in the amount of **structuring** and **annotation**

# Core focus: Symbolic music files

- **Research-grade symbolic music files** are surprisingly difficult to access
- Most existing scholarly music repositories focus on references to physical sources, to images or audio recordings
  - Many repositories do not reference symbolic music files at all
  - Most of those that do reference symbolic music typically:
    - Treat them as an afterthought, rather than as valuable digital objects worthy of careful consideration
    - Neglect essential issues like provenance and documentation of essential editorial and encoding decisions that are fundamental to conducting proper computational musicological research
    - Limit the range of symbolic formats available, contrary to the needs of researchers who in practice will need music available in a range of different formats (and who know that naïve automatic translation can bias or otherwise compromise research results)

# Metadata and feature searches

- SIMSSA DB may be searched using traditional metadata queries:
  - **Free-text** search
  - **Faceted** metadata filters, such as:
    - Contributor
      - Composer, arranger, author of text, transcriber, etc.
    - Instruments / voices
    - Sacred / secular
    - Genre / type of work
      - e.g. madrigal, motet, etc.
    - Etc.
- SIMSSA DB also permits **content-based searches** based on **features**

# Wait, what is a “feature?”

- Information that **measures a characteristic** of a segment of music in a **simple, consistent** and **precisely-defined** way
- Represented using **numbers**
  - Can be a single value, or can be a set of related values (e.g., a vector of histogram bin values)
- Provides a **summary description** of the characteristic being measured
  - Usually provides a **macro** rather than local view
- Usually extracted from pieces or distinct sections (e.g., mass movements) **in their entirety**
  - But can also be extracted from smaller segments of music

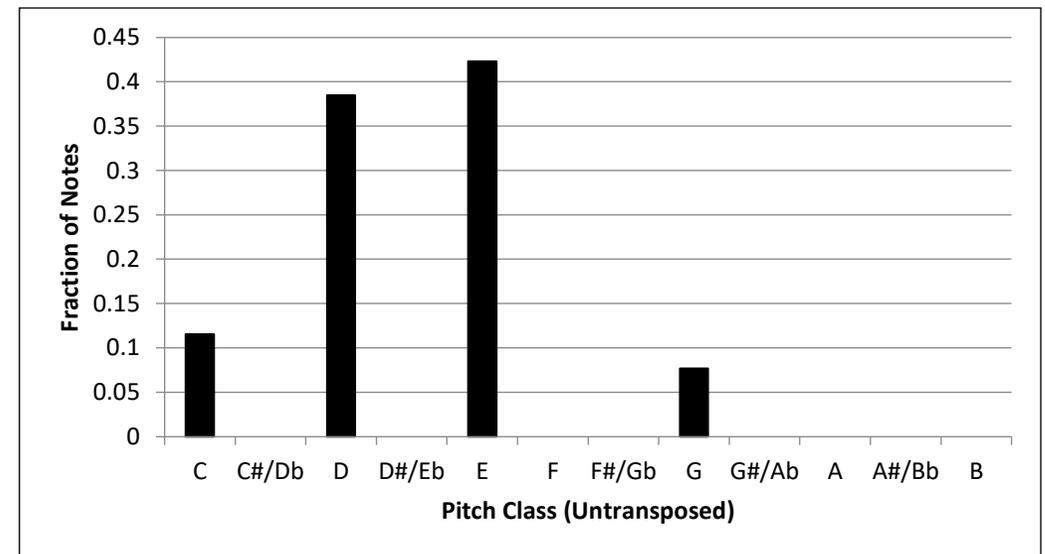


# Example: A histogram feature

- **Pitch Class Histogram:** Consists of 12 values, each representing the fraction of all notes belonging to an enharmonic pitch class

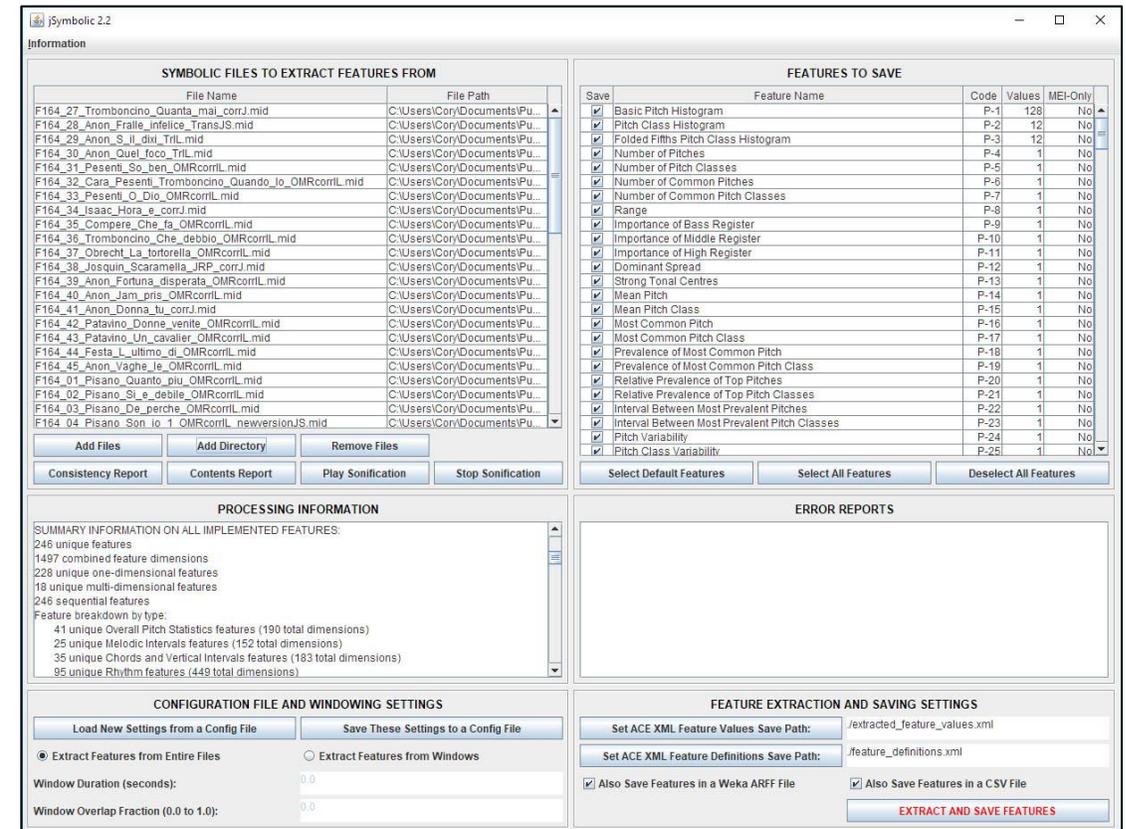


- Histogram graph on right shows feature values
- Pitch class counts:
  - C: 3, D: 10, E: 11, G: 2
- Most common note is E:
  - 11/26 notes
  - Corresponds to a feature value of 0.423 for E



# How might one calculate features?

- The **jSymbolic** research software (McKay et al. 2018) can be used to automatically extract features from **symbolic digital scores**
  - Open source
  - Applicable to diverse musics
- Version 2.2 extracts **246 unique features**
  - 1497 separate feature values, since many features a multi-dimensional (e.g. histogram vectors)
- The upcoming Version 3 extracts 533 unique features
  - 2040 feature values, including **n-gram features**



# jSymbolic 2.2's feature types

- Pitch statistics
  - e.g. Range
- Melody / horizontal intervals
  - e.g. Most Common Melodic Interval
- Chords / vertical intervals
  - e.g. Vertical Minor Third Prevalence
- Texture
  - e.g. Parallel Motion
- Rhythm
  - e.g. Note Density per Quarter Note
- Instrumentation
  - e.g. Note Prevalence of Unpitched Instruments
- Dynamics
  - e.g. Variation of Dynamics

The screenshot displays the jSymbolic 2.2 application window, which is divided into several functional panels:

- Information Panel:** Contains a table titled "SYMBOLIC FILES TO EXTRACT FEATURES FROM" with columns for "File Name" and "File Path". It lists 24 files, including various MIDI files like "F164\_27\_Tromboncino\_Quanta\_mai\_corrJ.mid" and "F164\_04\_Pisano\_Son\_lo\_1\_OMRcorrl\_newversionJSJ.mid".
- Features to Save Panel:** A table titled "FEATURES TO SAVE" with columns for "Save", "Feature Name", "Code", "Values", and "MEI-Only". It lists 25 features such as "Basic Pitch Histogram", "Pitch Class Histogram", "Range", and "Pitch Class Variability".
- Processing Information Panel:** Provides a "SUMMARY INFORMATION ON ALL IMPLEMENTED FEATURES:" including statistics like "246 unique features", "1497 combined feature dimensions", and a breakdown by type (e.g., "41 unique Overall Pitch Statistics features").
- Configuration File and Windowing Settings Panel:** Includes options to "Load New Settings from a Config File" or "Save These Settings to a Config File", and radio buttons for "Extract Features from Entire Files" (selected) or "Extract Features from Windows". It also has input fields for "Window Duration (seconds):" and "Window Overlap Fraction (0.0 to 1.0):".
- Feature Extraction and Saving Settings Panel:** Contains text boxes for "Set ACE XML Feature Values Save Path:" and "Set ACE XML Feature Definitions Save Path:", checkboxes for "Also Save Features in a Weka ARFF File" and "Also Save Features in a CSV File", and a prominent "EXTRACT AND SAVE FEATURES" button.

# SIMSSA DB and features (1/2)

- jSymbolic 2.2 has been integrated into the SIMSSA DB
  - Whenever an extractable file is uploaded to the SIMSSA DB, **features are automatically pre-extracted**, stored and indexed
- Users can specify **feature-range queries** via a **slider** for each feature they are interested in

The image displays the SIMSSA DB feature selection interface. On the left, a vertical list of feature categories is shown, including Chords and Vertical Interval Features, Dynamics Features, Instrumentation Features, Melodic Interval Features, Musical Texture Features, Pitch Statistics Features, Rhythm Features, and Rhythm and Tempo Features. On the right, a detailed view of the 'Melodic Interval Features' category is shown, featuring several sliders and numerical ranges for specific features:

- Average Interval Spanned by Melodic Arcs: 3.493 - 7.286
- Average Length of Melodic Arcs: 1.365 - 3.667
- Chromatic Motion: 0.0922 - 0.3582
- Direction of Melodic Motion: 0.3333 - 0.5909
- Mean Melodic Interval: 1.597 - 3.057

# SIMSSA DB and features (2/2)

- Can also **download complete feature sets** directly and use them as input to statistical analysis and machine learning tools (or analyze them manually)
- Feature searches can also be **combined with metadata searches**
  - e.g. retrieve all sacred pieces attributed to Josquin that contain parallel fifths

# Sample query combining metadata and features

The screenshot displays a search interface for musical works. On the left, there are several filter sections: 'Search' with the input 'amor', 'Sort By' set to 'Best Match', 'Composition Year From' and 'To' fields, 'Genre (Type of Work)' with checkboxes for Madrigal(8) and Frottola(1), 'Genre (Style)' with Renaissance(9), 'Composer' with Festa, Sebastiano(4), Pisano, Bernardo(4), and Tromboncino, Bartolomeo(1), 'Instrument/Voice' with Voice(9), and 'Sacred or Secular' with Secular(9). A 'File Format' section is partially visible at the bottom.

The main results area shows '9 Musical Works for query "amor" and selected facets'. A blue button 'Add Search Results to Cart' is present. The first result is 'Amore amor quando io speravo' by Pisano, Bernardo (1490--1548), categorized as Madrigal (Renaissance). Below this, there are four file format options: xml, midi, pdf, and sibelius, each with a green plus icon. The second result is 'Che deggio far che mi consigli Amore? [2, Pisano, F&H]' by Pisano, Bernardo (1490--1548). The third result is 'Hor vedi Amore che giovinetta donna' by Pisano, Bernardo (1490--1548).

On the right side, there is a note: 'Please note that features only apply to valid MIDI, Music XML and MEI files, and will exclude file formats from Sibelius, Finale, etc. For an explanation of all features, please consult the JSymbolic Manual.' Below this is a vertical list of feature categories: 'Chords and Vertical Interval Features', 'Dynamics Features', 'Instrumentation Features', 'Melodic Interval Features', and 'Musical Texture Features' (which is highlighted with a blue border). Under 'Musical Texture Features', there are three metrics: 'Average Number of Independent Voices' (1 - 3.938), 'Contrary Motion' (0.079 - 0.2071), and 'Maximum Number of Independent Voices' (1 - 4). Each metric has a small square icon and a horizontal slider below it.

# Abstract works, sections and parts (1/2)

- The SIMSSA DB maintains a conceptual separation between **abstract musical works** and **particular instantiations of them** (as expressed by particular symbolic files, for example)
- Multiple versions of the same abstract work can exist, and these should be both **associated with** and **differentiated from** one another
  - e.g. different editions, arrangements, etc. of a work
  - e.g. different digital symbolic encodings of the same manuscript
    - Could be in different formats (e.g., MIDI vs. MEI vs. MusicXML vs. kern vs. mscx)
    - Two versions could also use the same format, but be encoded differently (e.g., approach to *musica ficta*, base rhythmic note values, etc.)

# Abstract works, sections and parts (2/2)

- The SIMSSA DB makes it possible to divide music into abstract **Works, Sections and Parts**
  - Symbolic files sometimes contain whole pieces, and sometimes only subsets of pieces
- The makes it possible to **keep track of complex abstract relationships**
  - e.g., a single movement of a mass might be reused in another mass
  - e.g., an orchestral score and a keyboard reduction of it have different parts, but they are also different versions of the same abstract work

# Sources and provenance

- Keeping a record of **provenance** is musicologically essential
- Each digital object in the SIMSSA DB (e.g., a symbolic music file) is therefore linked to a **Source**
  - A “source” is a **reference** (ideally a URI) to a **physical or digital** document from which a digital object in the SIMSSA DB (e.g., a Music XML file) was derived
- Each source can in turn be linked to its parent source(s) through (eventually) **chains of provenance**
  - e.g., an MEI file transcribed from a printed score, derived from a hand-written copyist’s manuscript, derived from a hand-written original manuscript in the composer’s hand

# Other aspects of the SIMSSA DB

- Authority control
- Links to corpora
- Links to specific experimental studies
- Links to other types of data (text, audio, images, etc.)

# “Music researchers jumping into a lake” (according to Fotor and Deep AI)



# Basic metadata fields we would like LinkedMusic to make searchable and accessible

- Basic metadata fields
  - Title of work (including variants)
  - Title of section (including variants)
    - e.g., of a mass movement
  - Composer, author of text, arranger, transcriber, performer, improviser
    - With attribution certainty
  - Instrument / voice
  - Sacred or secular
  - Genre (Type of Work)
    - e.g., Motet
  - Genre (Style)
    - e.g., Renaissance
  - Dates
    - Precise or ranges
  - Locations
  - Source of item
  - File format
    - MusicXML, MEI, MIDI, kern, Sibelius, PDF, etc.

# Trickier things we would like LinkedMusic to make searchable and accessible

- Relationships and structures between entities
  - Work <-> Section <-> Part
  - Chains of provenance
  - Corpora found in
  - Experimental studies used in
  - Connections between related:
    - Symbolic music files
    - Audio files
    - Image files
    - Text files
- Feature values

The SIMSSA DB, Cantus DB, The Session and MusicBrainz are serving as early “music research test subjects”



# Some tricky issues to think about

- How will fields with no easy Wikidata property mappings be made accessible?
  - e.g., jSymbolic3's 2040 feature values
  - Having each repository provide formally structured URI documentation for each of potentially hundreds or thousands of fields would be onerous, but not intractable
  - Alternatives, like basic unstructured text search for such situations?
- Musicologists have repeatedly stressed the essential value of having the option to enter free text values (when they need to)
  - As opposed to being exclusively limited to options suggested by authority controls, that have existing URIs associated with them
    - e.g. titles or names not recognized by authority controls, or qualifications about dates
  - Is auto-creating new corresponding URIs in such situations a good option?
    - Probably not ideal from the perspective of reconciliation
- Similar issues likely relevant to other LinkedMusic repositories too

# Credit to the deserving

- I (Cory) designed the original data model and provided high-level guidance to the project, along with **Julie Cumming** and **Emily Hopkins**
  - **Andrew Hankinson** helped get us started with insights from DIAMM and elsewhere
  - Informal discussions with a range of musicologists have also been immensely valuable in guiding priorities
- **Ichiro Fujinaga** generously hosted SIMSSA DB development in his lab
  - **Gustavo Polins Pedro** and **Yaolong Ju** implemented the first version
  - **Rebecca Mizrahi** recently resurrected the DB implemented substantial improvements
  - **Hong Van Pham** has worked on deployment and towards LinkedMusic integration

# Live demo by Rebecca Mizrahi

- Staging version of SIMSSA DB:
  - [db.staging.simssa.ca](http://db.staging.simssa.ca)
  - May only be accessed via the McGill network
  - New contribution submission is currently **enabled** for data security
  - Please try it while you're here
- Production version of SIMSSA DB:
  - [db.simssa.ca](http://db.simssa.ca)
  - Publicly accessible
  - New contribution submission is currently **disabled** for data security

# Thanks for your attention!

[cory.mckay@mail.mcgill.ca](mailto:cory.mckay@mail.mcgill.ca)

[rebecca.mizrahi@mail.mcgill.ca](mailto:rebecca.mizrahi@mail.mcgill.ca)

[van.pham2@mcgill.ca](mailto:van.pham2@mcgill.ca)

