INDRA: A VIRTUAL SCORE PLATFORM FOR NETWORKED MUSICAL PERFORMANCE

Drake Andersen
Vassar College
Poughkeepsie, NY, USA
dandersen@vassar.edu

ABSTRACT

This paper introduces Indra, a Max-based virtual score platform for networked musical performance. Indra allows a conductor to improvise with an ensemble over a local area network by determining the notation that appears on performers' screens in real time. Musical compositions for the Indra platform consist of short encoded or image-based notation clips that are tagged with customizable metadata corresponding to musical qualities. The conductor uses the metadata to filter clips, determining the general qualities of the musical texture while the software cycles through clips that meet the filter criteria. Indra is designed to be accessible and adaptable for musicians working in diverse styles and numerous performance contexts.

1. INTRODUCTION

Real-time music notation systems, also known as virtual scores, have been used by an increasing number of composers over the past two decades to explore novel computer-based performance paradigms. Virtual score software by creators like Nick Didkovsky, Jason Freeman, and Georg Hajdu invite composers, performers and listeners to engage with notation in new ways, whether improvising with animated graphic notations, modifying notation on the fly, or even undertaking "extreme sight reading" [1, 2].

Individual virtual score platforms exhibit diverse functionality, occupying a spectrum from software designed for the performance of an individual composition, such as Jason Freeman's *Glimmer* (2004) or Craig Vear's *On Junitaki Falls* (2017), to systems designed for a wide variety of uses with minimal stylistic or aesthetic constraints. One of the most important and influential general-purpose systems is Georg Hajdu's *Quintet.net*, first developed in 1999, which allows musicians at up to five different locations to perform together over the Internet [3]. More recent platforms for networked notation display include Decibel ScorePlayer [4], DrawSocket [5], and SmartVox [6]. In addition to their respective specialties, each of these

platforms grapples effectively with the challenges of synchronizing notation across a network. However, virtual scores are also an opportunity to explore alternative, more flexible models for the selection and distribution of notation in performance.

Indra is a Max-based virtual score platform for networked musical performance that allows a conductor to improvise with an ensemble of any size by determining the notation that appears on performers' screens in real time.[1] Unlike many other virtual score platforms, Indra is intended to support a wide range of aesthetic and stylistic possibilities through its flexible interface, robust notation support, and customizable metadata and tagging systems.

Indra is designed to be accessible and adaptable so that musicians without previous experience can quickly learn to use the software and organize performances independently. You can download the latest version of the Indra software and documentation from this website: **https://creativeinteraction.org**

2. OVERVIEW

The Indra platform comprises three Max patches: one each for the composer, conductor, and performers. Since Indra is a platform and not a composition in and of itself, it can be used for the performance of many different compositions, as well as multiple performances of the same composition. Compositions in Indra are called *collections*, and can be generated individually or collectively. Existing musical works can also easily be imported and remixed using Indra. There is no functional limit to the number of performers.

The Indra workflow is deeply collaborative. The composer(s) create(s) a collection of short musical ideas or passages called *clips* that are passed along to a conductor, who interprets and arranges them during the performance alongside the musicians in the ensemble. Even though much of the material originates with the composer, the conductor and musicians shape the form, texture, and feel of each performance. Many performances with Indra incorporate improvisation, graphic notation, and indeterminacy, further distributing creative agency.

2.1 Performance Workflow

Collections are first created using the composer patch and saved as a database file that can be loaded into the other
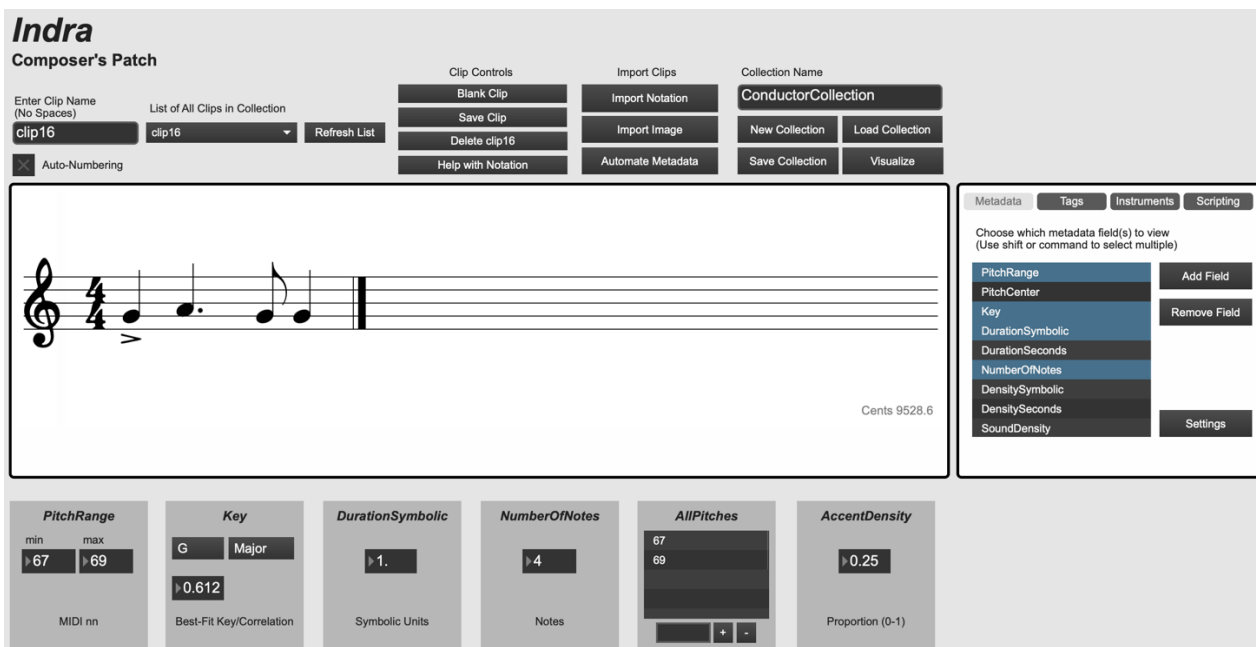
---

[1] https://cycling74.com/

Figure 1. The composer patch during the creation of a collection. Selecting metadata fields from the list on the right dynamically populates the metadata field boxes below the notation display. The metadata field boxes are used by the composer to display and modify metadata corresponding to the clip currently being displayed.

two patches. Collections consist of short encoded or image-based notation clips that are tagged with customizable metadata corresponding to musical qualities. The conductor patch allows the conductor to connect to the performers over a local area network and send messages that determine the notation that appears on the performers' screens. The conductor filters the clips using their metadata to filter clips, determining the general qualities of the musical texture while the software continuously cycles through clips that meet the filter criteria. The primary function of the performer patch is to display notation to the performer in response to the conductor's messages.

Many existing virtual score systems are designed for expert users, and present significant obstacles to musicians without experience using specialized music software. While this is often mitigated through the direct intervention and participation of the creator, these obstacles sometimes discourage would-be users from engaging with the software. Consequently, a central design priority for Indra is ensuring a smooth workflow for musicians through a clear and consistent operational logic.

2.2 Aesthetic Background

Indra was conceived in dialogue with a variety of existing performance practices, from the open scores of Anthony Braxton and Earle Brown to the meta-language systems of John Zorn and Walter Thompson. Indra reflects a fundamentally nonlinear conception of composition, meaning that individual sounds and gestures are organized into short clips designed to be rearranged to shape musical meaning. Clips may use traditional or graphic notation,

and may be divided by instrument or available to multiple performers. It is also possible to conduct nonlinear remixes of existing compositions, such as a Beethoven string quartet, or Stravinsky's *Symphonies of Wind Instruments*, by dividing them into clips.

2.2 Technical Description

Indra operates in the Max environment and makes use of the bach, cage, dada, odot, and zero packages, as well as several externals and abstractions distributed with the patch. Performances with Indra are conducted over a local-area network using UDP messages, employing the zero-configuration networking framework. Musical compositions for the Indra platform are stored as native bach *.llll files consisting of encoded notation and metadata, organized as a relational database. Compositions that employ image-based notation are distributed with an accompanying "images" folder that contains all image files.

3. NOTATION

Notation is created and modified using the composer patch. Figure 1 gives the composer patch during the process of creating a collection. Notation in Indra can be stored in a collection in one of two ways: (1) encoded as a bach *.llll file, or (2) an image file that is displayed to the performer. Indra uses the bach suite of objects [7] developed by Andrea Agostini and Daniele Ghisi to store and display encoded notation.[2] Notation for clips can be generated within Indra using the [*bach.score*] object graphic user interface, or by sending scripting messages through a text-based interface in the software. Among several Max-

---

[2]The official bach site: https://www.bachproject.net/

based notation frameworks, including MaxScore [8], bach was selected to be the basis for encoded notation in Indra due to the robust tools for analysis and visualization in the sibling cage [9] and dada [10] packages.

Users can also generate notation in the software of their choice and import it into Indra using the MusicXML format. Indra parses imported MusicXML files into clips by interpreting bars containing only whole rests as separators. Batch naming and tagging is supported. The use of encoded notation also allows for the automatic generation of metadata, as described below.

Indra supports the use of image files to represent clip notation as well. Batch importing, naming, and tagging of images is supported. Using images instead of encoded notation ensures the consistent appearance of the notation and allows for the use of graphic, colored, multi-staff, nonstandard, or otherwise unsupported notational elements. One drawback, however, is that at present all metadata information for image-based clips must be entered manually.

4. METADATA AND TAGS

Indra uses a relational database structure to store three parallel layers of information about each clip: (1) metadata, (2) tags, and (3) instrument tags. This data is used by the conductor to filter clips during the performance, but remains invisible to performers.

4.1 Metadata

Metadata refers to the information stored with a clip that describes its musical qualities, including pitch content, key, duration, density, and many others. In the current version of Indra there are fifteen default metadata fields: PitchRange, PitchCenter, Key, DurationSymbolic, DurationSeconds, NumberOfNotes, DensitySymbolic, DensitySeconds, SoundDensity, AllPitches, AllPitchClasses, DurationType, DurationRange, StaccatoDensity, and AccentDensity.

Each of the default metadata fields is user-definable, but can also be generated automatically for programmatic notation using analytical tools, several of which rely on the cage and data sibling packages to bach. For example, the Key field is determined through an implementation of the Krumhansl-Schmuckler key-finding algorithm. Indra supports customizable major and minor profiles, and also outputs a correlation coefficient that can be filtered by the conductor for distinguishing between the perceived strength of a key between clips.

Several of the default metadata fields are considered *functional* metadata, meaning that they have an effect on the operation of the software in performance. For example, the DurationSymbolic and DurationSeconds fields are used to determine the length of time a clip is displayed on a performer's screen. Similarly, PitchRange data can be used to silently filter clips by instrumental range. If this setting is turned on in the conductor patch, only clips whose PitchRange values fall within the standard instrumental range of a given instrument can ever be assigned to that instrument.

In addition, there are fifteen slots for custom metadata fields whose values and formats are entirely user-definable. Custom metadata fields can take one of six formats: (1) a single integer, (2) a single floating-point number, (3) a minimum-maximum pair of integers, (4) a state from several composer-definable options, (5) a list of words and/or numbers, and (6) the special compound key format which can be used to define scales, modes, set classes, collections, and other key-like data types. Users can also specify the type of control surface the conductor will use to operate the filter in performance, including sliders, notation-based displays, keyboard-based displays, menus, and lists.

4.2 Tags and Instrument Tags

The tagging system functions in parallel with the metadata system, but independently and with somewhat different functionality. Tags are custom labels that are stored as part of a collection and can be applied to any clip. A clip can be associated with any number of tags. Figure 2 gives the tagging interface in the composer patch.
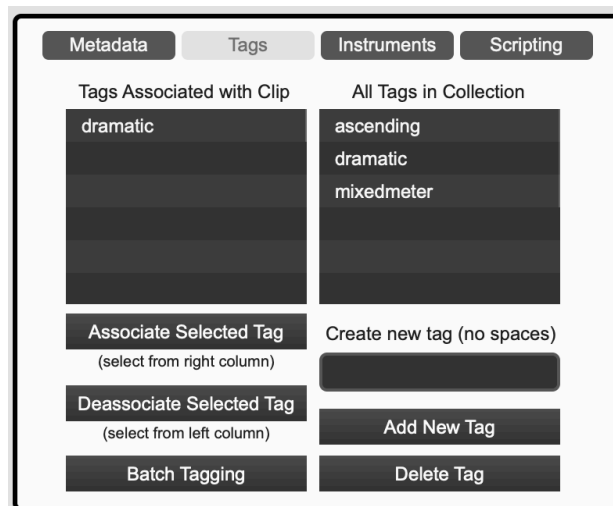

Figure 2. The composer patch tagging interface.

In addition to the standard tagging system, Indra also supports a separate tagging system specifically for instrumentation, which allows the conductor to automatically direct clips to particular instruments in performance. Instrument tags are similar to regular tags, except that they operate invisibly during performance. While performing a piece with Indra, the conductor must manually select which metadata and (regular) tags are active filters at any given moment. These settings generate a list of matching clips that are subsequently passed to the instruments. Instrument tags divide up this list so that only clips tagged for a particular instrument are actually sent to that instrument.

The conductor can utilize or ignore instrument tags as desired. Unlike regular tags, instrument tags are invisible: they do not appear in the main list of tags in the conductor patch, and once turned on operate automatically. Invisible operation speeds up the conductor's workflow by allowing the conductor to send instrument-specific clips to multiple
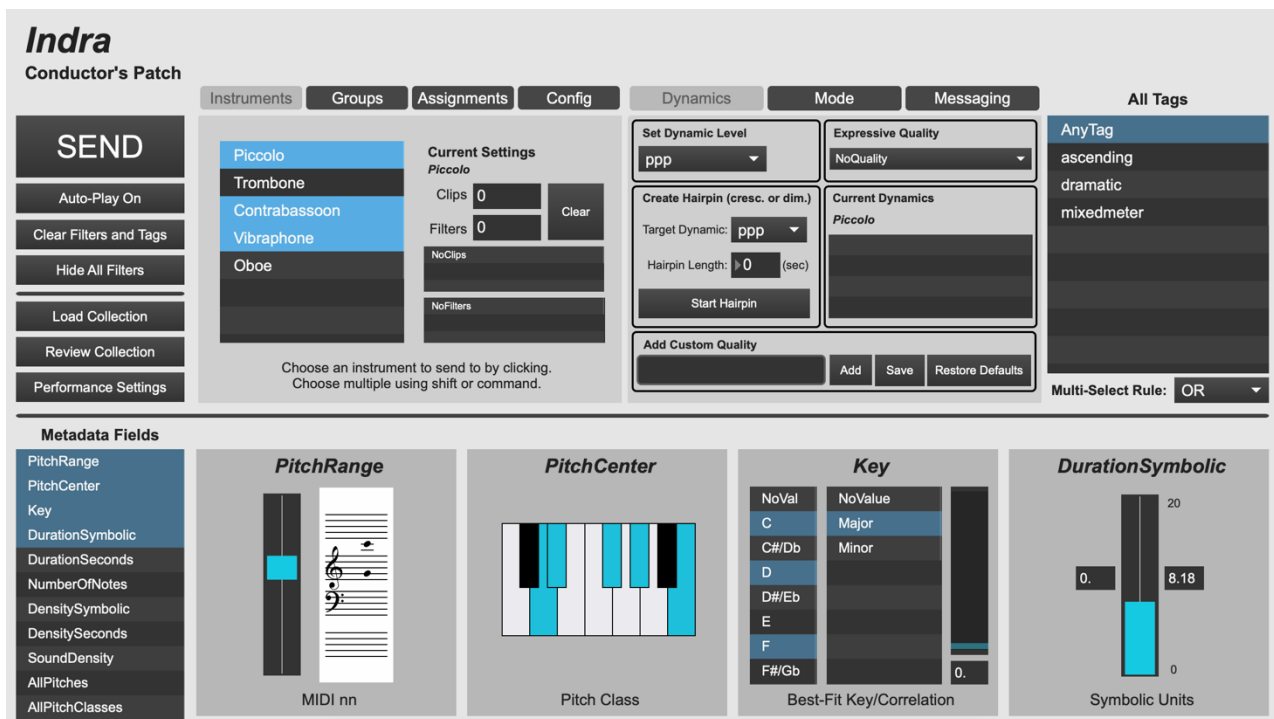
Figure 3. The conductor patch during a typical performance. As in the composer patch, the metadata field boxes appear dynamically based on the current selection in the Metadata Fields list on the left side of the screen. Unselected fields are automatically bypassed when sending new filter settings to performers.

instruments in a single operation. It also obviates the (otherwise redundant) need to select both a recipient and that recipient's instrument-specific tag. As with regular tags, a single clip can be associated with multiple instrument tags. Batch tagging is also available for instrument tags.

5. REHEARSAL AND PERFORMANCE

Indra is designed to make rehearsing and performance simple and straightforward for ensembles. Once the composer has completed the collection, the collection is distributed to the conductor and ensemble members for rehearsal and performance. The conductor leads the performance using their patch.

5.1 Filtering Clips

Figure 3 gives the conductor's interface as it might appear during a typical performance. The usual workflow is as follows. First, the conductor selects metadata fields from the Metadata Fields list in the lower-left corner. The corresponding metadata fields boxes will automatically appear in the lower half of the screen. The conductor then modifies filter settings using the interface elements in the metadata filter boxes. (The use of multiple filters selects for clips that meet all criteria.)

Next, the conductor may choose to restrict the clip selection to those with certain tags using the All Tags list in the upper right. The conductor can specify multiple tags using *AND* or *OR* logic. Finally, the conductor selects recipients by instrument or (user-definable) group from the multifunction box in the upper half of the patch on the left, and

presses the large send button in the upper left corner of the patch.

This workflow order is flexible: recipients can be selected before or after modifying filter settings. Clicking the send button will send whatever tags and filter boxes are visible on screen to whichever recipients are selected.

5.2 Performance Modes and Dynamics

There are three modes to which a performer can be assigned during a performance: play, tacet, and improvise. Play mode is the usual mode of operation, in which clips are automatically cycled through the performer's display, while tacet and improvise clear the notation display. Conductors who wish to use the improvise message are encouraged to discuss how to interpret this instruction with performers during the rehearsal process.

The Dynamics tab allows the conductor to send dynamics and expression-related messages to performers. Conductors may send one of three types of dynamics or expression-related information: (1) a static dynamic level (from *ppp* to *fff*), (2) a transition from one dynamic level to another over a specified amount of time, or (3) an expression indication. Expression indications may be chosen from a list of common indications such as "rising and falling in waves" or "irregular sf," or may be customized by the user.

The messaging system allows the conductor and performers to send instant text messages to one another through the Indra system. Intended primarily for troubleshooting in real time, the messaging system supports communication

from the conductor to one or more performers, or from one performer to the conductor.

6. DESIGN PRIORITIES

Indra prioritizes stability, adaptability, and a consistent operational logic so as to be as accessible as possible for musicians who are new to virtual score software. Supporting both encoded and image-based notation reflects a commitment to engaging with composers using traditional and graphic notation. The MusicXML import feature likewise allows composers to use the hardware or software of their choice to generate notation, rather than requiring them to learn and use Indra's built-in tools. This also facilitates collective and/or collaborative compositional practices.

Composers can take advantage of the customizable metadata and tagging systems to shape their performances. For example, besides strictly musical qualities, tags can also be used for theme groups, formal sections, and emotional affects. Finally, composers can use the [*dada.cartesian*] object to visualize their collection in progress by mapping default or custom metadata fields to different axes, or color, size or shape, and plotting clips as points.

The network configuration process—often the most tedious part of rehearsals and performances for musicians using networked notation systems—has been streamlined over several iterations into a fast and intuitive process through the use of the zero (Zero-configuration networking) package. This is especially true for the performer experience. Figure 4 gives the welcome screen that greets performers when they open the performer patch. Instead of having to deal with IP addresses or other technical information directly, performers simply enter a personal identifier (usually their first name and last initial) and their instrument, and then announce themselves over the network when prompted by the conductor. The zero package automatically resolves network addresses, populating a list in the conductor's patch with names and instruments.

Both the composer and conductor patches use dynamic interfaces for viewing and modifying metadata field values and filter settings, respectively. This keeps both interfaces as uncluttered as possible, while also making visually clear which filters are active and which are bypassed in the conductor patch during performance. In order to adapt to different conducting styles and performance contexts, the conductor patch settings can be saved separately from a given collection. For example, a conductor can save multiple rosters of performers in multiple settings files, but use the same collection for performances with each. Conductors can also assign and store custom groupings of performers, and select groups during performance using a single click, as demonstrated in the close-ups of the conductor patch given in Figures 5 and 6.

The simplicity of the performer patch, given in Figure 7, is intended to resemble traditional sheet music. At the same time, several features have been added as a direct result of previous performance experiences, including a preview window displaying the next clip to appear, an optional reference pitch for vocalists, a button to skip the current clip, a button to change clefs, and a flashing indicator to accompany the countdown as each clip advances. In addition, the performer patch includes a practice mode so that musicians can load a collection and cycle through clips on their own.

Figure 4. The welcome screen in the performer patch.

Figure 5. The Assignments tab in the conductor patch.

Figure 6. The Groups tab in the conductor patch. Clicking on the checkbox to the left of a group name selects all instruments within that group as recipients for the next message to be sent.

Figure 7. The performer patch during a typical performance.

7. CONCLUSIONS AND FUTURE WORKS

Indra is an accessible and adaptable virtual score platform for musicians working in diverse styles and numerous performance contexts. While Indra was originally conceived for concert settings, the composer Robert McClure recently used Indra to perform the musical accompaniment to a work for dance. Likewise, the author has organized performances using Indra that combine acoustic and electroacoustic forces, such as the chamber concerto for viola and ensemble *Spring Flow* premiered by Kallie Ciechomski. Upcoming workshops and performances focus on the creative exchange that takes place when composers conduct each other's works.

The next step in the development process will be incorporating gesture recognition technology into the conductor's interface for more intuitive and efficient control of the software. Additional areas for future development include using optical music recognition to automatically generate metadata from imported image files, and producing more extensive multimedia documentation and example collections**.**

Acknowledgments

8. REFERENCES

[1] J. Freeman. "Extreme Sight-Reading, Mediated Expression, and Audience Participation: Real-Time Music Notation in Live Performance." *Computer Music Journal,* vol. 32, no. 3, pp. 25–41, 2008.

[2] C. Vear. *The Digital Score: Musicianship, Creativity and Innovation.* Routledge, 2019.

[3] G. Hajdu, K. Niggemann, A. Siska, and A. Szigetvari. "Notation in the Context of *Quintet.net* Projects." *Contemporary Music Review,* vol. 29. no. 1, pp. 39–53, 2010.

[4] C. Hope, L. Vickery. "The Decibel Scoreplayer: A Digital Tool for Reading Graphic Notation." *Proceedings of the International Conference on Technologies for Music Notation and Representation (TENOR'15)*, Paris, France, 2015.

[5] R. Gottfried, G. Hajdu. "DrawSocket: A Browser Based System for Networked Score Display." *Proceedings of the International Conference on Technologies for Music Notation and Representation (TENOR'19)*, Melbourne, Australia, 2019.

[6] J. Bell, B. Matuszewski. "SmartVox – A Web-Based Distributed Media Player as Notation Tool for Choral Practices." *Proceedings of the International Conference on Technologies for Music Notation and Representation (TENOR'17)*, A Coruña, Spain, 2017.

[7] A. Agostini, D. Ghisi, "A Max Library for Musical Notation and Computer-Aided Composition." *Computer Music Journal*, vol. 39, no. 2, pp. 11–27, 2015.

[8] G. Hajdu, N. Didkovsky. "MaxScore: Recent Developments." *Proceedings of the International Conference on Technologies for Music Notation and Representation (TENOR'18)*, Montreal, Canada, 2018.

[9] A. Agostini, E. Daubresse, and D. Ghisi, "*cage*: a high-level library for real-time computer-aided composition", *Proceedings of the International Computer Music Conference (ICMC'14)*, Athens, Greece, 2014.

[10] D. Ghisi, C. Agon, "Real-Time Corpus-Based Concatenative Synthesis for Symbolic Notation", *Proceedings of the International Conference on Technologies for Music Notation and Representation (TENOR'16)*, Cambridge, UK, 2016.