

# And We Did It Our Way

## A Case for Crowdsourcing in a Digital Library for Musicology

David Bainbridge

University of Waikato, Hamilton, New Zealand  
davidb@cs.waikato.ac.nz

### ABSTRACT

This article makes the case for a digital library based on a crowdsourcing approach for musicology. At its heart, the argument draws upon ideas present in the popular music video TV show *Pop-Up Video*, a format devised in the late 1990s that embellishes the shown content with *info nuggets* that popup as bubbles and then disappear, as the video plays. We updated and extended the concept to operate in a web environment, choosing a digital library framework as a way to organize the set of videos contained in the site, and casting the popup information collated and displayed as metadata— aspects that further progress the argument for the developed software architecture being fit-for-purpose as a tool for musicologists. The article presents a walkthrough of the developed site, and then goes on to show how the elements present—particularly the gamification elements that focus on symbolic note content entered through a range of virtual musical instruments: piano, drum-kit and guitar—can be re-purposed for use by musicology scholars.

### CCS Concepts

•Applied computing → Digital libraries and archives;

### Keywords

Digital Music Library, Crowdsourcing, Music Visualization, Musicology Analysis

## 1. INTRODUCTION

When Frank Sinatra recorded the song *My Way*<sup>1</sup> in 1968, he sang a song about someone nearing the end of their life (‘And so I face the final curtain’) reflecting on the decisions they had made (‘I’ll state my case, of which I’m certain’), realizing that they were overall proud of the decisions they

<sup>1</sup>A song written by Paul Anka, in turn inspired by the melody of *Comme d’habitude* (As Usual) by Claude François and Jacques Revaux written in 1967.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DLJ/M ’15, June 24 2015, Knoxville, TN, USA

© 2015 Copyright held by the owner/author(s). Publication rights licensed to ACM. ISBN 978-1-4503-3563-8/15/06...\$15.00

DOI: <http://dx.doi.org/10.1145/2785527.2785529>

had made as an individual, difficult though it had been at times (‘And I did it my way’). Inverting the individual aspect of the song’s sentiment, tongue in cheek—“And We Did It Our Way”—in this article we make the case for a crowdsourced approach to supporting musicological study within the framework of a digital library. More specifically, by leveraging HTML5 [12] and in particular the Web Audio API [13], we illustrate how an intentionally frivolous pop music video website we have developed for entertainment, can be re-purposed to form a collaborative online environment in which musicologist can study music.

The article starts by detailing the original inspiration for the work described here (Section 2). This centers on the *Pop-Up Video* TV show developed by the music station VH-1, in which music videos are shown embellished with details that popup in-situ in the video. In Section 3 we present a walkthrough of *Pop-Up Videos Respoiled*, the digital library site we have developed that rekindles the ideas of the original show, but also updates and extends the ideas to operation in a web environment. For instance, in addition to the popup idea, we also implemented a play-along features akin to Guitar Hero.<sup>2</sup> Section 4 provides implementation details. Through the implementation process we realized that much of the functionality being developed was equally applicable to audio on its own, and further that much of what had been implemented could be re-purposed for use by musicologists. We make the case for this in Section 5, before concluding the article in Section 6.

## 2. BACKGROUND

The TV show *Pop-Up Video* first aired on VH-1 in 1996, and ran until 2002. In its heyday it was the station’s most popular show. The show later had a short reprieve with new episodes commissioned and aired 2011–2012. Figure 1 shows a snapshot taken from one of the show’s videos, featuring an *info nugget* (as they became known)—in this case, positioned in the lower portion of the shot. Popups appear roughly every 10–15 seconds as a video plays, and the location a popup appears varies through the course of a song, positioned where best fits the piece of information being displayed. For instance, this might be beside the instrument a band member is playing if the information displayed pertains to that. The sources of information used typically came from interviews with the film crews who recorded the videos: from the director to the sound-boom operator, and the artists themselves where possible.

<sup>2</sup>[http://en.wikipedia.org/wiki/Guitar\\_Hero](http://en.wikipedia.org/wiki/Guitar_Hero)



**Figure 1: An example Pop-Up Video (*Land Down Under* by Men at Work) from the VH-1 TV show of the same name.**

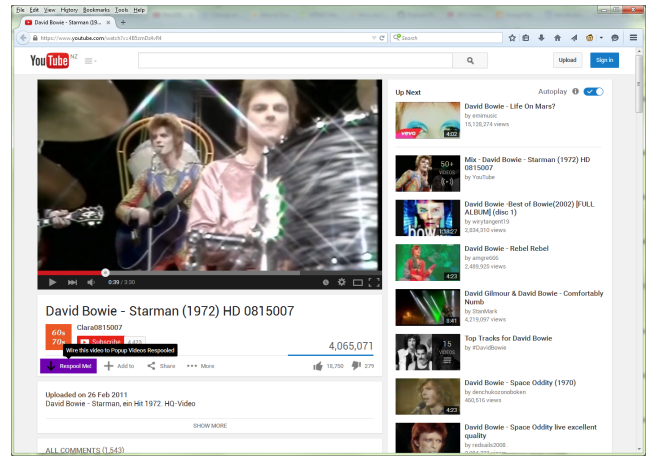
The initial goal of our project was to revitalize the popup video idea, but updated for a web environment. Using HTML5 and related web technologies [12], time-based popups can now be symbolically represented and associated with the underlying video footage—they no longer need to be “baked” into the video using post-production editing tools, as in the original format, with many advantages as a result. The approach, for instance, allows for more than one layer of popup information to be associated with a video. Moreover, the user can be given control as to which layer of information is shown, changing between individual layers, or else selecting multiple layers to be shown at the same time.

We also liked the idea of expanding the features provided by the site to (hopefully) increase the level of engagement users have with the site through gamification. The key idea here was to make use of the Web Audio API [13] to provide a play along capability (using virtual instruments the site provides such as piano, drums and guitar) that allows the song being played to sound full-bodied when the user is playing the correct notes, and faint and tinny when not.

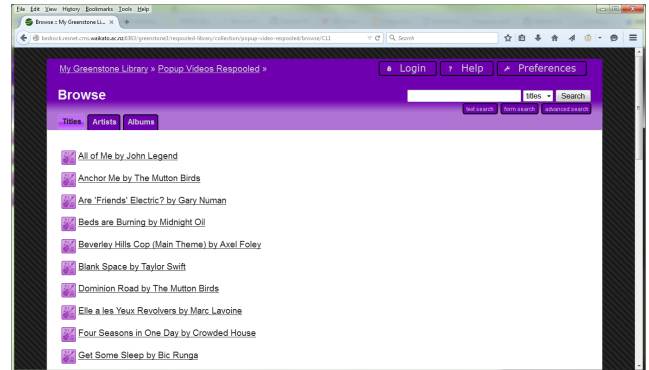
To support such gamification, this led to the requirement that the site needed to include symbolic information representing the notes being played in the pop video (ground-truth data, if you will). We resolved this issue by dual-purposing the virtual instruments provided by the site. In addition to the play-along feature, we also developed a recording mode whereby the notes (drum beat, or guitar chords) being played were captured and associated with a song. Further, an editor was included to allow the user making the recording to correct and embellish the captured data, as necessary. We will see the result of such crowdsourced data later in the article, in particular in Figure 10 of the walkthrough, and later in Figure 14 when the user-contributed data is repurposed for musicological study.

Finally, to manage the set of songs available through the site, we utilized a digital library framework. This opened up the opportunity to allow the popup information to be stored as metadata in the digital library, allowing the site to provide a search facility, essentially for free.

We called the developed website *Popup Videos Respoled* in honor of the origins of the idea. However, as we developed the site, we became aware that much of what we were doing—which had an increasing edutainment flavor—was equally applicable to more serious scholarly use. We return and expand upon this point in Section 5.



**Figure 2: Selecting a video from YouTube to feature in the *Popup Videos Respoled* web site.**



**Figure 3: Browsing by titles in the *Popup Videos Respoled* web site.**

### 3. WALKTHROUGH

The interaction experience of a user of *Popup Videos Respoled* starts when they are accessing mainstream video sites, such as YouTube and Google Videos. Through a browser extension to the user’s web browser (previously installed), when they visit a page in such a site, the web page being viewed is dynamically adjusted to include a *Respool Me!* button. An example of this can be seen in Figure 2, which shows the user playing a YouTube video for David Bowie’s *Starman*. Below the main video player area (hard up against the left of the page) is the *Respool Me!* button (purple background, to match the styling of the *Respoled* site). Pressing this results in the video ID being transmitted to the digital library providing the *Popup Videos Respoled* site. In Figure 3 we see the user has shifted their attention to the *Popup Videos Respoled* site, accessing in this instance the browser by titles page to access the song that is of interest to them.

Figure 4 shows the initial view the user is presented with when they select a song (incorporating a modest amount of interaction by the user, where they have set the video to play with the Popup Trivia layer enabled). The snapshot has been taken 22 seconds into the video of *Starman* by David Bowie, where the Info Nugget reads:

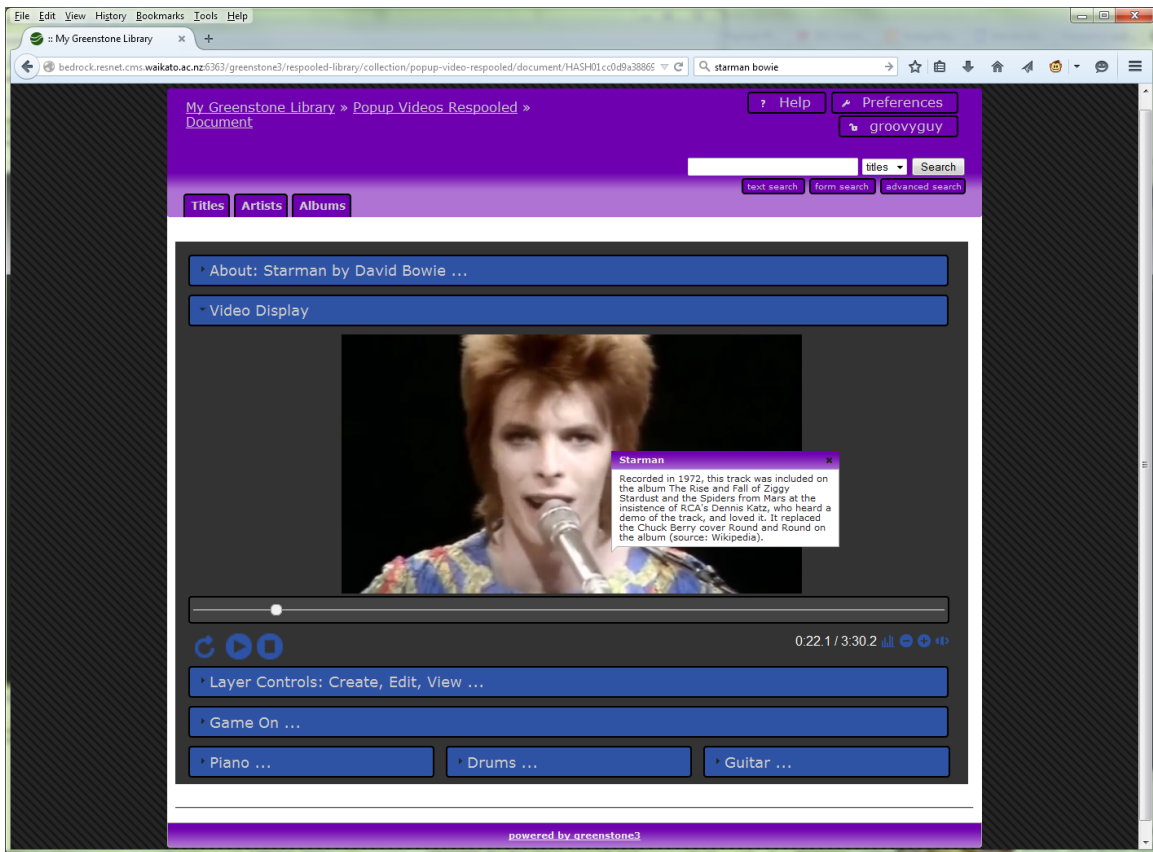


Figure 4: Initial document view for *Starman* by David Bowie, which also shows the recreated core popup feature in action.

Recorded in 1972, this track was included on the album *The Rise and Fall of Ziggy Stardust and the Spiders from Mars* at the insistence of RCA's Dennis Katz, who heard a demo of the track, and loved it. It replaced the Chuck Berry cover *Round and Round* on the album (source: Wikipedia).

Central to understanding how the interface operates is through its use of turnstyle bars. Notice how the web page is broken into various sections: About, Layer Controls, Video Display, Game On, Piano, Drums, and Guitar. These appear as turnstyle bars (blue in the figure) that exhibit a toggling open/close behavior when clicked on. Click on it once, it opens in a smooth animated style, showing an additional part of the web page; click again and it closes. Figure 5 shows the turnstiles in action, where the user has chosen to close the video area and open the "About" area to access further information about the song.

More than one turnstyle can be left open at a time, if desired. We omit showing this here in the figures to conserve space. The purpose of the other turnstyle areas are detailed in the following sections.

## Layer controls

Figure 6 shows the Layers Control section of the interface. This is divided into three areas: Popup Layers, Video Player Mode, and Play Along Layers. The Popup Layers area is the most readily understandable. It is where the user controls

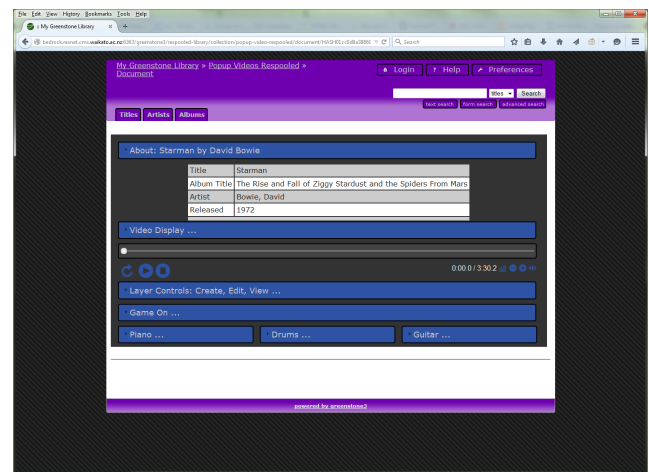


Figure 5: Accessing the content of the About turnstyle bar.

the popup layers that are shown when the video is played. This is the section of the interface the user interacted with briefly before the snapshot in Figure 4 was taken.

If a user is logged in (in our scenario we have a user, *groovyguy*, with an interest in 1970s music) with an account with the necessary permissions to edit the content associ-

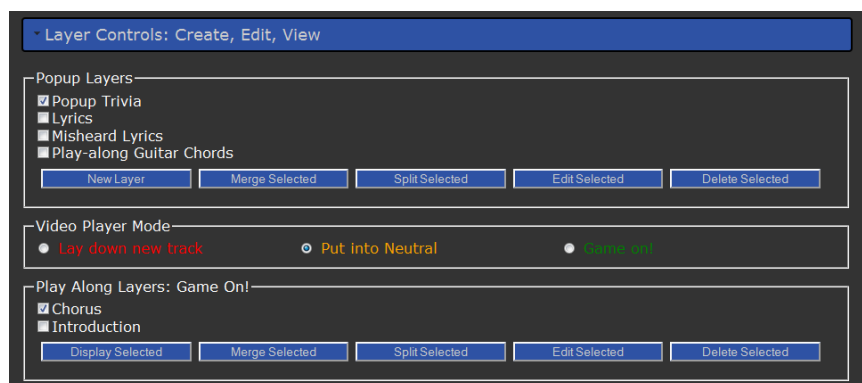


Figure 6: The Layer Controls section of the interface.

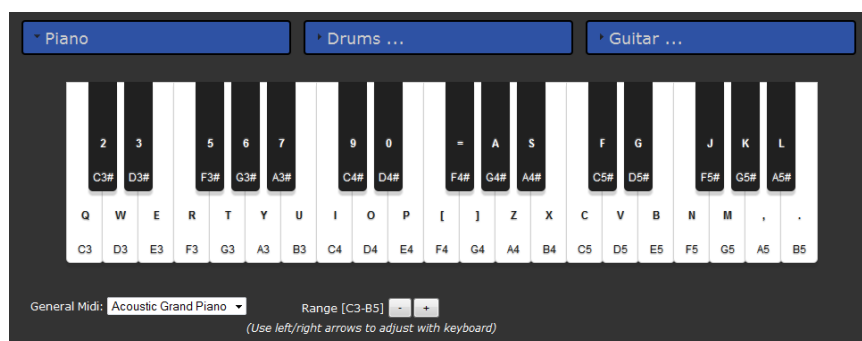


Figure 7: The virtual piano available for playing and recording pitched noted tracks.

ated with the song (they are in our case), then below the check-boxes for the layers to show, there is a row of buttons that support editing operations: new, merge, split, edit and delete.

The second and third areas of the Layer Controls section operate in conjunction with the Game On and virtual instruments (Piano, Drums, Guitar), and are related to the gamification aspect to the site. This is an aspect of the interface that will become clearer when we describe these latter turnstyle bars, however in brief:

- For the second area when *put into neutral*, nothing special happens when playing the video.
- If set to *lay down a track* when the video is played, any notes played on any of the virtual instruments are recorded (pitch and note-on and note-off data). On pressing stop, a popup window appears asking the user to name the track just recorded.
- If set to *game on* when the video is played then the audio to the video sounds tinny until the user starts playing the correct notes in time to the music using the virtual instruments. To assist in playing the pitch and timing of the notes, a piano-roll style visualization is available in the Game On turnstyle (described in more detail below).

The third area—Play Along Layers—controls how the (symbolic note-event) tracks that have been recorded are used. Check-boxes control which tracks are shown in the piano-roll format in the Game On section of the interface. Similar

to the Popup Layers, if the user is logged in with an account that has edit rights, then a row of buttons is available for editing the recorded track data.

## Recording a track

Figures 7–9 show the range of virtual instruments provided by the site: piano, drums, and guitar respectively.

The most natural way to interact with these instruments is through a multi-touch screen, a design decision that taps into the increasing prevalence of tablet devices, and that web browsers exist for each of the main mobile operating systems that support the Web Audio API. The virtual instruments can also be operated using mouse input reasonably effectively. QWERTY keyboard input is also available, which is fairly straightforward to master after a bit of practice.

## Play along overlays

Figure 10 shows the opened-up Game On area. This is the area where the piano-roll style visualization of tracks is displayed. Multiple tracks can be displayed at the same time, as controlled by the check-boxes in the Layers Control area. When the video is played, the piano-roll display starts to scroll vertically. The animation is timed so that the note-events depicted as rectangles in the visualization hit the red horizontal play-line as they become active. If the Control Layers area is set to “Game On” then the audio to the video played will only resonate in full when the notes in the piano-roll that cross the play-line are played on one of the virtual instruments.

In terms of determining the notes to play when recording

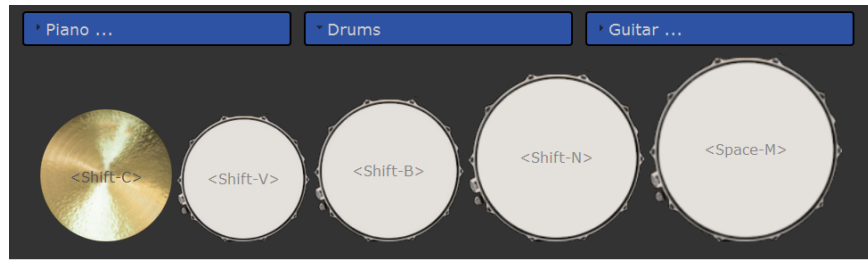


Figure 8: The virtual drum-set available for playing and recording atonal percussion.

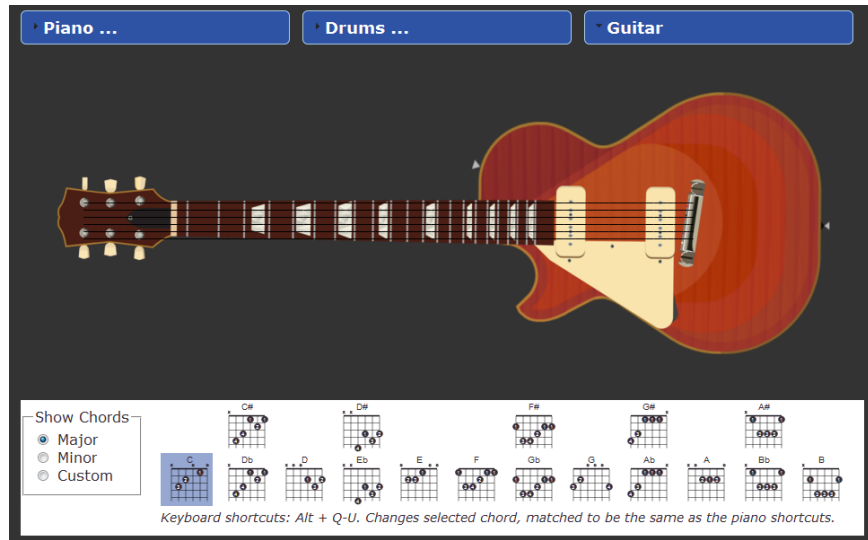


Figure 9: The virtual guitar available for playing and recording guitar strumming.

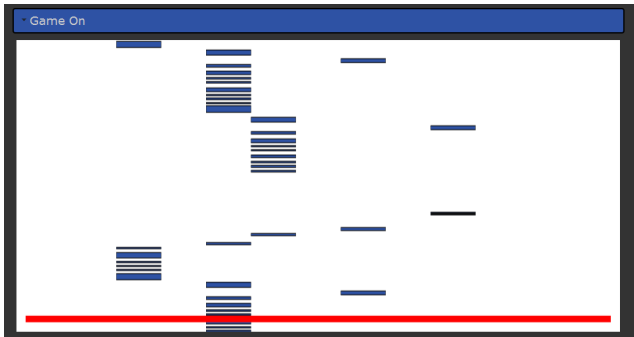


Figure 10: The “Game On” features in action, which is reminiscent of Guitar Hero, only in this case involving a virtual piano.

a track, it can sometime be difficult to hear all the nuances to the instrumentation and vocalization of a piece of music. For this reason the interface supports a graphic equalizer, left and right channels, as shown in in Figure 11 for filtering the audio. The control for switching the graphics equalizer on and off is located next to the volume controls below the video progress bar (see Figure 4).

#### 4. IMPLEMENTATION

In terms of implementation details, the heavy lifting to

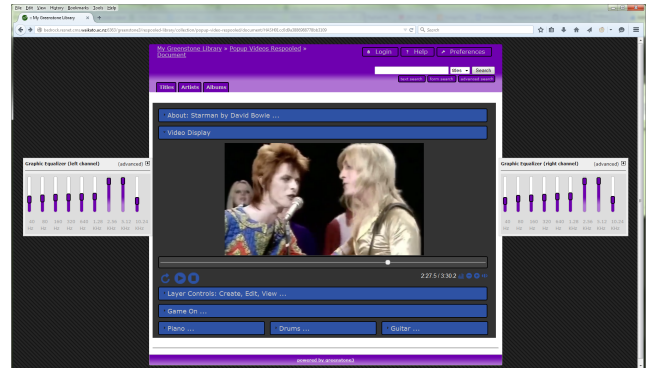


Figure 11: The “Game On” features in action, which is reminiscent of Guitar Hero, only in this case involving a virtual piano.

this project is accomplished using the Greenstone 3 digital library toolkit [15], an environment which the author is well versed in developing experimental web prototypes in. Nothing in the core digital library software had to be changed to develop the site. The rendering pipeline to Greenstone 3 is XSL Template (XSLT) based [14]. In addition to defining new XSL Templates to override default behavior in the digital library, the lion’s share of the implementation work to provide the desired *Popup Videos Respoled* functionality

was through Javascript code. This in turn drew upon a variety of Javascript libraries. In particular: the virtual piano is based on the Javascript library AudioSynth;<sup>3</sup> the drum kit on JQuery drum-set;<sup>4</sup> and the guitar based on details provided by Fred Chasen, one of the developers of the Jam on Chrome experiment.<sup>5</sup>

The browser extension mentioned in Section 3 that allows users to select videos from mainstream video sites to be listed under the *Respoiled* site was developed as a User Script [11]. This is a technique that allows developers to write extensions in Javascript that operate outside of the traditional security sandbox Javascript runs in, in a browser. A user installing a User Script is presented with a list of abilities the extension makes use of, and they must confirm acceptance of these details before the script is installed. For our work we make use of a User Script's ability to make XMLHttpRequest calls (aka AJAX calls) that are cross-domain. This is needed so the ID of the video being selected can be transmitted to the *Respoiled* site.

The quiet, tinny effect that can be heard when video plays in Game On mode, when the user is not playing the correct notes (or playing at all) is implemented using a `ScriptProcessorNode()`, an element of the Web Audio API. The effect is provided using a high-pass filter. The node passes through the input values to the output unaffected when the notes are being played correctly. There is a deliberate timing delay before the high-pass filter takes effect, to smooth over any minor differences in the timing of the notes played by the user, and what has been recorded in a given track.

## 5. A CASE FOR MUSICOLOGY

As outlined in the introduction, the more the development of the *Respoiled* site progressed, the clearer it became that what was being produced—with some minor adjustments—would be equally of interest to others, such as musicologists in supporting their studies. Now that we have detailed a walkthrough of the site as conceived for its original purpose, and further described how this was accomplished in terms of implementation details, we are now in a position to make the case for the described work being re-purposed to support crowdsourcing in a digital library for musicology.

First, clearly the digital library software is agnostic as to the video content it contains. While the catalyst for the developed work, there is nothing special about the pop music content with regard to this work. The input can be any type of video recording, say a performance of Richard Wagner's Opera *Die Walküre*, or a filmed recording of a choral performance of Benjamin Britten's *A Ceremony of Carols*. In either of these examples, the ability to overlay time-based information over the video is a useful annotation tool, especially as these annotations instantly become searchable within the digital library as a whole. The ability to have different layers of overlays would most likely come even more into its own when used this way, with different scholars developing different layers and/or contributing to layers started by others. While Greenstone 3 supports a fine-grained permissions model for controlling access to doc-

uments in a digital library, this does not currently extend to controlling access to user-supplied metadata, and so some implementation work would be necessary to fully realize this. In the case of orchestra performances, the fact that there is no sung (or spoken) words to comment on, and the visual spectacle is more static means the bubble annotation feature might seem superfluous; however, we believe there would still be a role for enriching such entries in the digital library with textual annotations this way. For instance, comments on what the conductor is doing at particular moments in time during the performance.

Next, the content need not be restricted to just video. Specifically, audio content is an equally obvious choice, when use as an online tool to support musicological study becomes the focus. From a technical standpoint, particularly well aligned with this is how HTML5 video and audio have been designed, through their shared use of the `HTMLMediaElement`. Given the turnstyle design, it would be a simple matter of omitting the Video Display block when the page being served is audio. In such a situation, the annotations could appear above the time-line progress bar, although we conjecture that it could very well be worth keeping the physical space that is presented when the video area is opened as a useful area in which to display annotations for audio. All that would be needed was a change in the terminology used to label the turnstyle-bar. Some user testing would help determine the effectiveness of this.

The development of such an envisioned website for supporting musicological study could also take advantage of other features the digital library software provides. For instance, one of Greenstone's strengths is that it can cope with a heterogeneous set of documents, in a wide variety of formats. Particularly germane here are text books formats, and scanned sheet music [1]. Multiple collections is another feature of the software that was unused in the *Popup Videos Respoiled* development, but would be extremely useful in this new use-case. In particular, it would allow scholars to develop specifically scoped worksets in which study could be undertaken.

Finally, Greenstone 3 provides a decentralized model to the editing of metadata, performed through a web browser by authenticated users. This editing of metadata is how all the popup annotations, and recorded tracks are managed in the *Respoiled* site, although this fact is masked by how the interface has been designed. Brought across to supporting musicologists in their academic pursuits, this editing ability naturally leads to a collaborative environment—a form of crowdsourcing.

### 5.1 A Modest Prototype

As an initial stepping-stone to providing the sort of environment described above, we have developed a (modest) prototype to demonstrate the direction we think the digital library functionality should be moving in. While the prototype reuses the same pop content (for the sake of expediency), it is easy to imagine the forms of music coming from genres such as classical, blues, or jazz.

Figure 12 shows the initial view a user sees when accessing an item in the reworked digital library collection. Comparable with the initial view of a item in *Respoiled* (Figure 4), there are several differences to note. First, the video area is not open by default, as in the new imagined scenario there will also be audio-only content in the collection, and so hav-

<sup>3</sup><http://keithwhor.github.io/audiosynth/>

<sup>4</sup><http://www.theonlytutorials.com/jquery-drum-sets-html5-audio-tag-example/>

<sup>5</sup><http://www.html5rocks.com/en/tutorials/casestudies/jamwithchrome-interaction/>



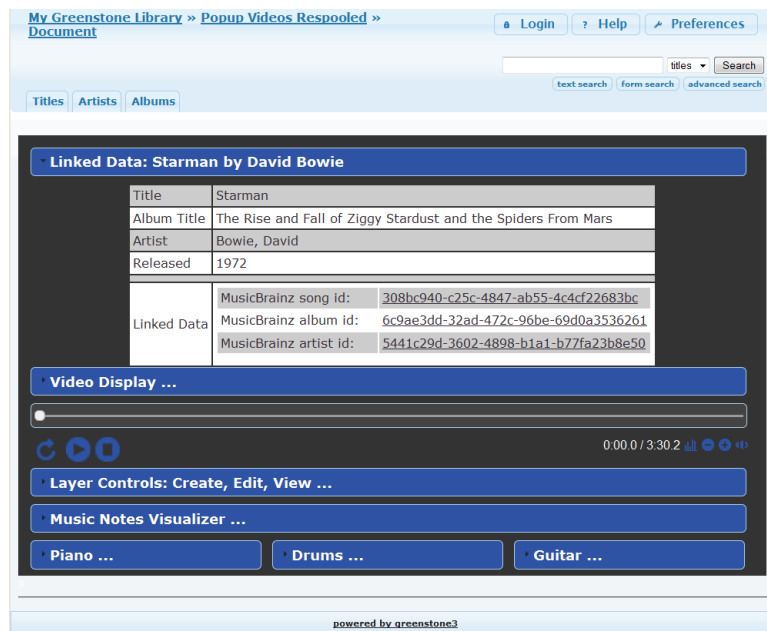


Figure 12: The Digital Libraries for Musicology version of the *Popup Videos Respoled* site.



Figure 13: A user entering the MusicBrainz ArtistId metadata for entry under study (Bowie's *Starman*) using Greenstone's browser-based, metadata editor supporting distributed, collaborative data entry.

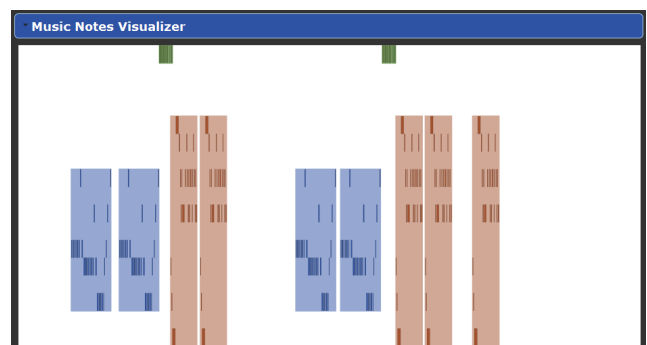


Figure 14: The piano-scroll view for “Game On” re-worked as a music note visualizer.

ing such a view open by default makes less sense. Rather, the first turnstyle-bar is open by default. We describe it as the first turnstyle-bar as this is the next change: the bar itself is now labeled “Linked Data: ...” and not “About: ...” Further, the information on display has been expanded, displaying MusicBrainz IDs—these have been entered by the scholar through Greenstone's browser-based metadata editor—covering a variety of forms associated with the item in the digital library on display: song ID, album ID, and artist ID. Figure 13 shows the user operating the editor to enter the relevant MusicBrainz's artist ID for *Starman*. Through customized XSLT rules, such additional metadata (if present) is presented to the user, automatically hyper-linked to the pertinent MusicBrainz's web page.

Figure 14 shows what used to be the *Game On* turnstyle area in the *Respoled* version. This is another area that has undergone numerous changes, starting with a name change to Music Notes Visualizer. It retains a similar piano-roll form of visualization, only now it has been rotated through 90 degrees, and scaled to provide an overview of the entire musical piece. When the video or audio is played, it remains

static, and it is the red time-line bar that moves from left to right as the performance progresses.

Another visual change is the blocking of sections to the tracks. These reflect the structures made when tracks are recorded using the site. For instance, when the chorus track to *Starman* was made, only the first occurrence of it was played. This appears as the fourth box in Figure 14. The first two boxes (rendered blue) are verses; then a small green box at the top, which is a pre-chorus element where the same single note in the song is rhythmically played before entering the chorus (shown in pink). Having recorded one rendition of the chorus, the editor was then used to clone a copy of it being played 9.8 seconds later (from when the first occurrence of the chorus started). Determining this numeric offset is a simple procedure using the timing information on display below the video's progress-bar to note when the two cycles of the chorus start.

If the figure, for illustrative purposes, this has also been done for some of the structural elements in the song. Ideally, though, this process should be repeated for all the verses,

choruses, bridge, intro and “outro.”

Not shown in the diagram is the interactive tooltip that appears when the user hovers over a block. This displays in the tooltip both the text label assigned to the block and the predicted key of the notes in the block. For the first verse in the figure, the tooltip reads:

Verse 1, Key: G (minor) with 56.7% confidence rating

The key detection (in the first instance) is automatically determined using the Krumhansl-Schmuckler key-finding algorithm [6]. Through the editing capabilities provided a scholar can override this if they wish, and further remove the confidence rating if deemed appropriate (or else set it to 100%).

## 6. CONCLUSIONS

In summary, this article has plotted a course of development from a digital library site that was only ever intended to be “a bit of fun” to one that has real potential as an on-line environment used by musicologist to collaborate in the pursuit of their scholarly studies. The intention here is to make the case that this is an area worthy of further study, by drawing attention to the possibilities.

A modest prototype encapsulating some of these ideas was presented. Utilizing a digital library framework naturally led to a collaborative environment in which scholars can crowdsource music information. However, given the metadata editing model provided by this class of software architecture, the specifics of the crowdsourcing interaction experience provided by the prototype was more pedestrian than we would have liked. To provide a richer and more sophisticated environment—to support, for example, the crowd-sourced input-agreement technique developed by Law and von Ahn for their *TagATune* Game With A Purpose [8]—the model for metadata editing used would need to be extended.

Further ideas, which draw upon research activities in the area of music content analysis, include:

- Enhance the visual capabilities through the displaying of a 12-bin windowed chromagram sequence [4] and self-similarity map [5, 3]. In the case of the chromagram sequence this could work well as an optional background to the music note visualizer.
- Enhance the search capabilities of the digital music library through Query by Humming (QBH) [10, 2]. The recorded note event tracks make an ideal source for this. Further, through the editor, blocks represent the main hook-lines to a song could be specially marked out.
- Add support for nested structure in the recorded tracks. Similar to the use of the Krumhansl-Schmuckler key-finding algorithm in the prototype to auto-populate recorded blocks with key information, the Repair algorithm [7] could be used to establish an initial attempt at the nested structure.
- Go beyond the use of virtual instruments, and allow input from physical instruments using the Web Audio API’s microphone support. Further, allow the playing of these instruments to be synchronized with audio content in the DL through score following [9].

Such capabilities would greatly enhance the experience of a musicologist utilizing the envisaged music digital library. However, we stress here that the technical aspects to this work are only a means to an end. The viability of the approach would literally become vacuous if musicologists did not engage with it to enrich the information stored in the digital music library. A partnership with a group of musicologists to develop the idea further is the next logical step, to fully realize the aspiration expressed in the title to this article: “And we did it our way.”

## 7. REFERENCES

- [1] D. Bainbridge and T. C. Bell. An AJAX-based digital music stand for greenstone. In *Proc. of the 2009 Joint Int. Conf. on Digital Libraries, JCDL 2009, Austin, TX, USA, June 15-19, 2009*, pages 463–464, 2009.
- [2] D. Bainbridge, M. Dewsnip, and I. H. Witten. Searching digital music libraries. *Information processing & management*, 41(1):41–56, 2005.
- [3] D. Bainbridge, X. Hu, and J. S. Downie. A musical progression with Greenstone: How music content analysis and linked data is helping redefine the boundaries to a music digital library. In *Proc. of the 1st Int. Workshop on Digital Libraries for Musicology, DLFm ’14*, pages 1–8, New York, NY, USA, 2014. ACM.
- [4] J. P. Bello and J. Pickens. A robust mid-level representation for harmonic content in music signals. In *Proc. of the Int. Conf. on Music Information Retrieval*, pages 304–311, 2005.
- [5] J. Foote. Visualizing music and audio using self-similarity. In *Proc. of the Seventh ACM Int. Conf. on Multimedia*, pages 77–80, New York, 1999.
- [6] C. L. Krumhansl. *Cognitive Foundations of Musical Pitch*. Oxford University Press, 1990.
- [7] N. J. Larsson and A. Moffat. Offline dictionary-based compression. In *Proc. of the Data Compression Conference*, pages 296–305, 1999.
- [8] E. Law and L. von Ahn. Input-agreement: A new mechanism for collecting data using human computation games. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems, CHI ’09*, pages 1197–1206, New York, NY, USA, 2009. ACM.
- [9] R. Macrae and S. Dixon. Accurate real-time windowed time warping. In *Int. Conf. on Music Information Retrieval*, pages 423–428, 2010.
- [10] B. Pardo, J. Shifrin, and W. Birmingham. Name that tune: A pilot study in finding a melody from a sung query. *J. of the American Soc. for Information Science and Technology*, 55(4):283–300, 2004.
- [11] M. Pilgrim. *Greasemonkey Hacks: Tips and Tools for Remixing the Web with Firefox*. O’Reilly Media, 2005.
- [12] M. Pilgrim. *HTML5: Up and Running*. O’Reilly Media, 2010.
- [13] B. Smus. *Web Audio API*. O’Reilly Media, 2013.
- [14] D. Tidwell. *XSLT: Mastering XML Transformations*. O’Reilly Media, 2nd edition, 2008.
- [15] I. H. Witten, D. Bainbridge, and D. M. Nichols. *How to Build a Digital Library*. Morgan Kaufmann, San Francisco, 2nd edition, 2009.