Preview Cues: Enhancing Access to Multimedia Content

m.c. schraefel, Max Wilson, Maria Karam
Electronics and Computer Science
University of Southampton
Southampton, UK
[mc, mlw101, amrk03r] @ ecs.soton.ac.uk

ABSTRACT
We describe preview cues, a lightweight mechanism to assist exploration of multimedia content. A preview cue provides a *preview* of the kind of content/information associated with an *area* (as opposed to an instance) of a domain. Preview cues associate media files and their meta data with the label of a topic in a domain. A lightweight gesture such as brushing a cursor over a label initiates playback of the preview cue file associated with that label. With these cues, users can preview the *type* of content associated with an area of a domain in order to decide whether or not that area is of interest for further exploration *before* having to select it. In this paper we describe the preview cues mechanism. We look at one case study of an implementation of preview cues in the audio domain, and we present the results of a user study of preview cue deployment. We conclude with a discussion of issues for future research.

KEYWORDS: Multimodal, audio interfaces, exploration

INTRODUCTION
We present preview cues, a lightweight multimodal interface mechanism designed to provide previews of an area or category within a multimedia domain. Preview cues can be used to assist users in determining whether or not an area of a domain is of interest for further exploration: preview cues let users quickly compare multiple areas of a domain in their current context, so that they do not have to select and explore each area individually to make an assessment about a category.

The development of preview cues has been motivated by the challenge of exploring multimedia content in real time when one is not particularly knowledgeable about a domain of interest. If we are looking for a video to rent or an album to buy we may find it difficult to wander into the less well known aisles of a store or areas of a web site to find something new, due to this lack of knowledge. In both the physical and the digital domains, we often rely on the familiar to leverage the less familiar: we look for a name we might recognize; we search out a title a friend has recommended. In other words, where domain knowledge is not great, we tend to rely on following a path based on a direction from trusted sources. Preview cues are designed as an alternate or complement to such recommendations. Preview cues foreground attributes of an area such as what it sounds, looks or behaves like. With this information, people can rapidly make determinations *on their own* about whether or not they are interested in pursuing content with those features.

In this paper, we describe the preview cues mechanism, and situate preview cues within related work. We present a case study in which we deploy preview cues, and we present the results of a user study. We then discuss future work with preview cues.

PREVIEW CUES
Preview cues are designed to help users quickly get a sense of areas of a domain by providing access to features of these areas so that, based on inspection of these, users can determine whether that area is of interest to them for further exploration, *before* having to explore it. This self-directed inspection of resources is what Cathy Marshall and Frank Shipman refer to "information triage" [13]: the rapid assessment of an information space to determine whether or not a source is of current interest or value. With preview cues, users with previously poor access to a domain can do something not previously well supported in either the physical or digital: instead of having to rely on recommendations, they have a mechanism whereby they can rapidly "triage" a domain for themselves. By providing multimodal cues about a domain area, we enhance the users' ability to perform information triage on a domain space by providing more forms of concurrent information from which users can make assessments and formulate decisions. This use of multimodality to improve access to and manipulation of information is captured by what Oviatt et al have referred to as the goal of multimodal interfaces to deliver "more transparent, flexible, efficient, and powerfully expressive means of human-computer interaction" [14].

Interaction

Preview cues have five attributes: (1) a topic label to which the preview cue file is associated, (2) a link associated with the topic label that, on selection, connects the user to more selections under that category, which themselves have associated preview cues, (3) the preview cue media file, (4) that file's associated metadata, and (5) collection space for local compilation of selected previews. There are also three gestures associated with preview cues: inspection, selection and addition.

Inspection. A preview cue is triggered by a lightweight gesture, such as brushing a cursor over a topic label. The gesture initiates the playback or streaming of a media file and the display of appropriate attributes of that file's metadata. For instance, brushing over the category label "film noir" would trigger both the playback of, say, the Maltese Falcon, and the display of information about the preview cue itself, such as the date the film was produced, and the list of its director, writer and stars – information which may help users further understand the attributes of a category. The Maltese Falcon is not the only film noir movie but it is part of the film noir category, and users can get a sense of what film noir is about from this cue and its associated metadata.

Selection. If users decide they are interested in the topic, selecting the topic label opens a list of labels associated with the next subcategory of the domain. In a film directory organized by Genre, Decade, Directors, Films, for instance, a user selecting the topic Film Noir under the category Genre, would open up a list of dates under the subcategory Decades. The user can again hover over the dates (40s 50s etc) to inspect preview cues of film noir movies associated with each decade, and so on.

Special Cases: Instance Cues, Addition and Collection. At the end of a listing of a domain, one can no longer open topics into further subtopics: only the instances of that final subcategory are left. In that case, preview cues become "instance cues" – the cue associated with the label has a one to one correlation between the label and the file associated with it. The other case in which preview cues become instance cues is when a preview cue is added to a collection. A play list in the preview cue sense is a collection of preview cues: at any time while previewing cues, users can add the currently playing cue to these collections. The motivation for these collections is to increase support for information triage: if users decide that a cue in a particular area is of interest to them, they can add it to their collection for later access. In this way, they can continue to explore the domain, confident that they can return to their selections for further exploration later.

Presentation

Preview cues are visualization agnostic. They could be implemented as links in lengthy documents where brushing over a link in the text triggers a cue and displays the textual information about the cue playing; clicking on the link to select that topic takes one to another document about that category in the domain, which is itself populated with references/links from that topic, or they could be associated with cone trees [16] for example. Fundamentally, preview cues assume some kind of hierarchical slice through an n-dimensional space from which a user can explore nodes of the given hierarchy. We present one visualization in the case study below.

CASE STUDY

Preview cues arose as part of a larger project to develop mSpace, a "domain browser" [19]. A domain browser brings associated content on a given domain together, so that the topic can be explored rapidly and easily from multiple perspectives. We have been inspired by the promise of the Semantic Web [1] to make heterogeneous content available for this kind of dynamic association. As part of this multimodal inspection of information in a domain space, we developed preview cues as a first phase inspection method to locate areas of interest in a domain. Once an area is selected, associated information about the selected topic is made available as a second level of inspection. In a Jazz domain browser, for instance, preview cues would help users first determine an area of interest, perhaps cool jazz. Once that area is selected, information about cool jazz becomes available, such as a text definition of the term, a video clip of an artist talking about cool jazz, and links to categorized online resources about cool jazz.

Our goal has been to make the domain browser available in a Web context, since this is where most people with access to networked computers do their information exploring. On the Web, most hierarchies are presented as one node/one page at a time, where path information is available as a set of links at the top of the page. We compared using preview cues with this temporal hierarchical interface against using preview cues with a simple, multicolumn page model (Figure 1, below). Not surprisingly, performance for selection tasks improved significantly with the spatial browser [18].

Based on these findings, we settled on a multicolumn view for exploring domain hierarchies. These can be implemented with Javascript in compliant Web browsers. We describe one such prototype in "CS AKTive Space: Representing Computer Science in the Semantic Web" [20]. That prototype let us focus on particular Semantic Web issues rather than interface-specific concerns. To test preview cues in this interface layout, however, we developed a Java application rather than a Web/HTML/JavaScript interface. The Java code was straightforward to instrument for testing without having to focus on browser-centric issues for the prototype. For storage, we used a Postgres database.

Domain

For our first prototype, we chose music rather than film for the domain. Given limited resources, we could build a larger collection of digital audio than digital video to use for preview cues. To determine which area in music to build, we first took a straw poll in our lab to propose candidate genres where the criteria was High Interest Low Access. By "low access" we meant that participants are interested in an area, but are not effectively able to search for content on their own. We then ran a survey with our poll's top two categories, Jazz and Classical, and asked 200 people online again to rank the two areas for High Interest/Low Access. Classical music won.

Once we settled on Classical, we then carried out a survey of music-oriented Web sites in general, and classical ones in particular. The survey revealed that few sites that represented an area of music, as opposed to a single artist, used audio to support descriptions of artists or genres. It is common for artist fan sites to associate short audio clips with specific tracks of albums in an artist's discography. In this case, artist sites presume an audience. They are already the "instance level" of a particular genre. As such, it is a short path from albums in the artist's discography to tracks, to associated audio samples with those tracks. The IBM Glass Engine project [10] is an exception to this model. In this interface, users move sliders labeled with names, dates, genres and other variables such as "sorrow" "joy" and "density" to explore Philip Glass's music. The audio played back for any setting represents the piece associated with that particular set of sliders' current position: there is a one-to-one correlation between the state of the sliders and a single work that results as an output of that combination.

In the case of sites representing larger music spaces than a single artist, access to content assumes a certain level of previous knowledge. In other words, the music provided only as an end point rather than as a guide. The Indiana University's Variations project [23] has digititized its library's audio collection. It is focused, however, on supporting library users such as students and researchers who already know the domain and therefore can use keyword and category searches to retrieve the music. Likewise in electronic music, Discogs, provides a thorough database listing of artists, labels and venues [6], but unless users know what they are looking for, the site's alphabetical listing is not particularly helpful. There is also no music associated with the site.

Epitonic, on the other hand, presents "cutting edge" content [7], in several ways. From their Radio page, users can select the number of genre categories of interest, and select the number of tracks they wish to hear in total. A customized stream of music matching the criteria then streams to the user's media player. Information about each track is displayed as it is played. There is no clean link however back to the site from a specific artist being streamed, and no way to store the information about a stream automatically. Beyond the Radio page, the site also has a preview-cue like feature: its "genre walkthrough" pages have playable links that stream samples of pieces from a genre while one of the site contributors talks over the track to explain that genre. While effective, the site has a high human maintenance cost: it is hand crafted by guest editors who assemble play lists, construct walkthroughs, and recommend new artists.

While Epitonic represents best practice for hand built music sites, online music stores such as Amazon.com or the Apple Music Store represent best practice for more automatically managed collections. Both stores use short hierarchies similar to single artist sites: Period |Album | Piece for classical or Performers | Albums | Piece for everything else, and likewise only provide music samples at the end of the hierarchy. Further, these sites are still largely designed around the physical music store model: potentially fine if one knows what one is looking for, but all the same difficulties for exploring the less known.

Preview cues deployment within the mSpace domain browser is an effort to bring some of the handcrafted effectiveness of Epitonic to the more automatically generated music sites like stores and libraries.

Prototype Interaction

In this section, we describe how the specific attributes of preview cues have been implemented in our case study.

Inspection via Hovering. Beside each category label is an icon of a speaker. Hovering the cursor over a label causes a dash to be displayed for cursor position feedback Hovering the cursor over a speaker icon triggers the playback of a media file associated with that label's category. Once a preview cue is triggered, the metadata about the preview cue is presented in the upper pane of the interface. Information includes the name of the work, as well as its associated path/categorization in the domain. Only hovering over another cue will cause the currently playing track to change. At any time, users can also press the stop button to stop the currently playing cue.

Selection via Clicking. Clicking on a category label has two effects: it selects the label, highlighting it. Selection also expands that label, populating the next column/category of the interface. Each time the user makes a selection that area of the path is highlighted, and remains highlighted until the user makes a different selection. Thus, the current path stands out from the context of available topics in a category, as shown in Figure 1, where the path is highlighted in blue.

Addition via Double Clicking. At any point in the exploration of the hierarchy, a user can double click on a selection to add the currently playing preview cue to a a collection (lower pane of Figure 1). Selected pieces in the

collection list can be auditioned by hovering the cursor over the label of the piece. Pieces can also be deleted by selecting them, and pressing the "delete from list" button.

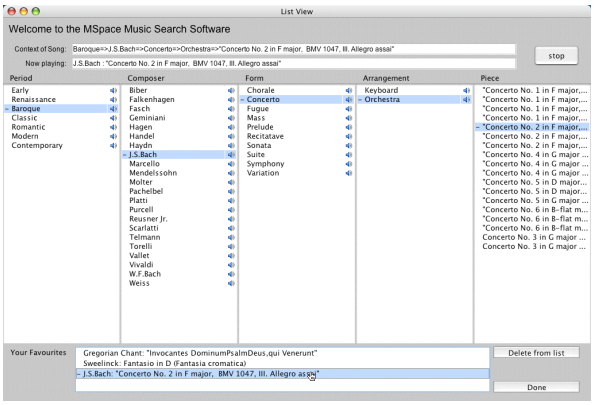


Figure 1. Multicolumn preview cue broswer for exploring a multimedia domain space.

Association of Cues with Labels

The hierarchy for classical music we use for the browser is based on music experts' categories for the domain: Period | Composer | Form (such as symphony or concerto) | Arrangement (e.g. violin, guitar) | Pieces (the instances themselves). Selection of available media files to associate with the labels in each category can be determined in at least three ways. First, they can be selected *randomly* from the database, constrained only to match a given category. Second, a *recommendation* list for each category of the domain can be established, and files selected against that list, in so far as the database has those pieces. Third, selections can potentially be determined *representationally* by algorithmically comparing the characteristics of the files available in a category and providing the cue for the most representational of that set.

In our case, we used a hybrid approach between recommendation and randomization. Sony Music Canada donated 70 albums for the project. Two classical music experts tasked to choose "top" works across periods compiled the list of 70. We then supplemented this set with an additional 90 albums. These were compiled from the top 100 classical recordings from the NPR guide to classical music [12], excluding the several opera selections from that list. This list was again reviewed by our experts, and supplemented with recommendations from the contemporary period in particular. We have 175 recommended albums in the database. Each album has associated metadata that was screen scraped from various media company web sites against the list we compiled of the recordings. The result gave us metadata about the recording, period, composer, form, arrangement, work, performer, conductor and piecel(title). We had to use scraping because, as we learned from Sony and Universal Canada, the record companies did not have such lists themselves. Their lists largely only included title, label and product code number for a specific recording.

Once we had the metadata assembled and the audio files stored, we developed a simple algorithm to randomly assign a preview cue audio file to a label. From the resulting 1490 tracks, the system selects a piece that matches the criteria of the currently active label. So, if the only constraint on a selection is Period: Romantic, then any of the files in the database matching Period: Romantic are eligible to be selected as cues. Once the selection criterion reaches the instance level of the hierarchy in the final column of the browser, there is at that point only one file eligible for matching the label.

Since our entire, if relatively small, collection is based on recommended works, users are only being exposed to a recommended/representative set, even when those tracks are randomly assigned to the labels. When the collection extends beyond Top of the Charts listings, other mechanisms may be needed for determining cues. We touch on some possible approaches here.

Recommendation, as one technique for candidate selection, has been well studied (see [9] for a survey of these) and has its own costs and benefits. One of the main costs, of course, is determining which recommendation approach to use and for what kind of recommendation: trusted community experts vs. musicologists vs. top sellers and so on. We are also interested in seeing if, with a large-scale collection of thousands of discrete recordings, there is any significant difference in performance or experience between using automatic random selection for preview cue assignment and using some kind of recommendation scheme.

Another approach to preview cue selection may eventually be able to be based on automatic determination of most representative piece from within a category in a collection. Work in music information retrieval has developed algorithms both for automatically detecting representative parts of an individual piece [3], and for retrieval of audio based on pre-specified criteria [8] [24]. We are not, however, aware of extant work that analyzes the discrete audio tracks available within a category and then determines the *most representative* piece of that set. The closest work in this area is automatic clustering of music content into categories based on feature detection and comparison (see [11] for a survey of these approaches). While success has been relatively low, new feature extraction techniques like those proposed by Li [11] have shown improvements in this space. Better classification results have been found with manipulations on midi rather than audio files [4] where transpositioned-normalized MIDIs and clustered into a binary tree. While effective, this technique is limited to the existence of available MIDI files across periods and categories of a music genre. In either case, it may be possible to find one

MIDI or audio file with a set of features whose total or average or median distance from all the others is the least, and thus, marking it as the representative instance.[1] To our knowledge, such an approach has not yet been developed.

Despite the feasibility or not of recommendation, randomization or representation, the question of best practice for determining preview cues from within collections, is an open one. While studies would need to be undertaken to determine whether or not there are significant differences between one approach and another, the approach designers use may be as much informed by the type of experiences they wish to provide users – community based recommendation; content based representation – as by whether random selection is just as effective or not as expert list recommendations.

RELATED WORK

Preview cues are related to a number of techniques for providing real-time information about the categorization and status of elements represented by an interface. In 1994, Pollitt, Ellis and Smith [15] presented their HIBROWSE interface for databases in which information in the database was first classified against a thesaurus, and then the number of documents matching each category was presented. New categories could be invoked to constrain the original list. So, the number of documents for a disease might be modified when the filter "drug therapy" is applied. The simple association of number of documents available when constraints are applied lets users know something about the domain space: in this case, whether or not there are documents which match their criteria (papers on drug therapy treatments for a specific disease) and how many of them their may be. In a similar vein, also in 94, Tweedie, Spence, Williams and Bhogal and Su introduced the Attribute Explorer in [22]. The explorer is a visual query style interface where attributes of a domain are presented with scalar values. Selection of a domain element causes histograms to be drawn against the values on the sliders. In this way, users can get a fast view of the distribution of the particular entity against a variety of criteria, and thus decide if that entity is appropriate for further investigation. In both the HIBROWSE and Attribute Explorer, users are given information to help them make decisions about whether or not an entity is of interest to them *before* they get to the entity itself.

Schmandt's Audio Hallway project [17] investigated another way to preview audio content before determining a selection or focus in an audio space. Rather than expanding hierarchies, the Audio Hallway presents clusters of audio "braids" which are associated by topic in particular audio spaces/rooms down a virtual hallway. In any part of the hallway, related audio news stories were "braided" so that while each story in a cluster played concurrently, each story was also temporarily foregrounded for the user by having it panned to the center and the volume raised. Head movements were used to select one strand in the rope braid rather than another.

Tooltips are invoked by brushing over an associated icon. They reveal the name of the command associated with the icon. They associated a *single* functions or *single* definitions of a term with a single label. Likewise, Brewster's work in earcons has been to use audio to provide specific cues to communicate the state of an interface or representation, whether that state be the degree of completion of a download [2, 5]. Preview cues do not provide the one to one correlation of an icon to a function description or function state. Instead, they provide candidates from *areas* that match domain criteria and thereby provide a sense of the kind of content which is part of that area of a domain. Similarly, Terry and Mynatt's Sideviews [21] provide previews for graphics applications in which an artist can preview the effects of multiple versions of a filter's settings on an image, rather than a seeing a preview of only one filter setting at a time to facilitate selection of the appropriate setting.

USER STUDY

The hypothesis motivating preview cues is that the presence of such cues increases the accessibility of a domain space in particular in areas where users have less domain knowledge than the terminology for the domain's representation presumes. Someone who does not know the difference between a serenade and a concerto may be hard pressed to choose between them, for instance.

Single or Multiple Cues

Our preliminary study [18] had shown that preview cues do enhance both user experience and task performance when users were asked to make selections from the domain, their core task being to build a play list of five classical music selections they would like to add to their own music collection. We had set up one treatment of the interface so that cues, in keeping with Web best practice, were available at the end of the hierarchy, and one treatment where preview cues were available at every level of the hierarchy. Preview cues throughout showed significant positive effects over cues only at the end of the hierarchy.

In that preliminary study, a single preview cue was associated with each label of the domain for the interface treatment where preview cues were persistently available. Reviewers, rather than the participants of the study, raised the concern that, by providing only one cue per label, we may not be fairly representing an area of a domain, and thus are prejudicing a user's assessment of

[1] The authors are grateful to Lloyd Rutledge of CWI's, Multimedia Group for suggesting this variation on clustering work for representational selection.

that area: one cue for an artist may not provide a person with enough information to decide whether or not they may potentially wish to write off that area of a domain, so a preview cue may *misinform* users about a category.

Our hypothesis was that we were not misrepresenting an area of a domain by associating a piece that was genuinely categorized as part of that domain. Likewise, most of the participants in our study went from previously having effectively *no* access to the classical music domain, to having significant access for making selections from it. The concern, however, that the single cue might cause a person to avoid an area of a domain if that sample were not in itself sufficiently indicative to represent the scope of an area, caused us to reconsider preview cue presentation. We extended preview cues to support multiple cues per label. Rather than presenting one icon for preview cue inspection, we present several, in an effort to give the user a broader sense of the music associated with a category.

The use of multiple preview cues is similar to the Audio Hallway audio braid in that it offers multiple instances within a topic/category. In our case, the user determines their focus by brush over of a given cue, rather than by head direction bringing up the volume of concurrent sample. The core difference between the audio braids and preview cues is context: preview cues are in place to support exploration of an organized domain. The Audio Hallway braids were used to represent all the instances themselves of a topic available in virtual region of the hallway, with no particular (apparent) domain organization to those topic regions. In other words, braids were shown to be effective for separating out instances within a topic, not for exploring organized domain spaces. Consequently, we wanted to evaluate whether or not something braid-like, in this case, multiple preview cues, provided any statistically significant performance improvement over single preview cues for domain exploration.

Just as there are several ways to determine files to associate with preview cues, there are several ways to determine the number of cues to offer for a given label: a preview cue for each subcategory of that label could be used, or some ratio based on the number of instances available for a category may be another mechanism. Simply providing a fixed number of cues per category is another. For this study, we decided to offer a fixed number, and this for two reasons. First, we settled on three cues per sample since that was the maximal range of cues that our small database could support. Second, using a fixed number of cues also gave us two consistent interface types for comparison. The single cue interface had one cue per label (Figure 1); the multicue version had three. No noise would enter the data about perceptual issues regarding varying numbers of cues per label in the multicue interface. Except for the number of cues that could be inspected, the two interface conditions were then the same. In each case, participants hovered a cursor over a speaker icon to the right of the category label to initiate a preview cue. In the case of the multicue version of the interface, three speaker icons were provided (Figure 2).

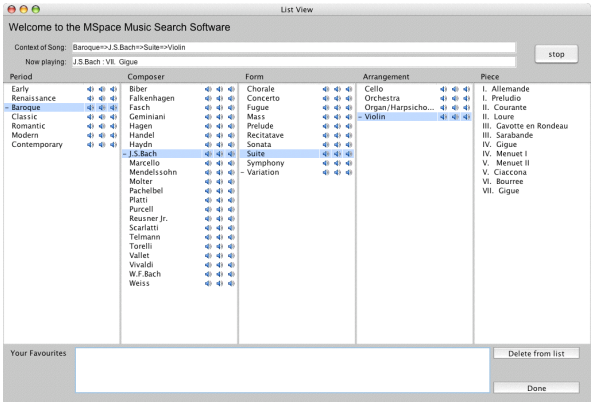
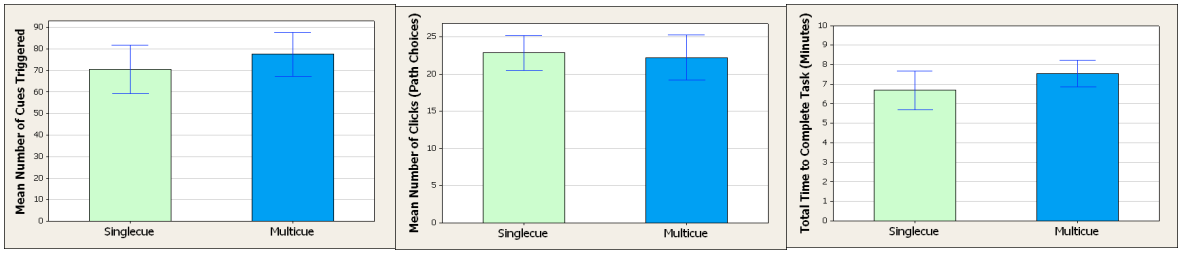


Figure 2. Multicolumn interface for multiple preview cues.

Study Method

To look at the difference between the use of single and multiple preview cues, we conducted two rounds of usability tests. In our first study we ran a between groups design where one group used the single preview cue browser and the second group used the multiple group browser. In the second study we ran the same protocol with a counter-balanced within group study. While both the within and between group studies measured the same quantitative and qualitative effects, we could focus more specifically on performance comparisons with the two browsers in the between groups, and subjective experiences comparing both approaches in the within group. Our hypothesis was that since most users had little knowledge about the classical music domain, one or many cues was not going to make a significant difference in their decision to explore down one path of the hierarchy or another. We did hypothesize however that path actions



(a) Mean cues by interface (b) Mean clicks by interface (c) Mean task time by interface

Figure 3. Different means for each interface. Error bars show ±1 standard error.

would be different: with more cues, participants would explore fewer paths as deeply, using the multiple cues to make earlier assessments about an area.

For the evaluation we used a task based on our earlier design study: participants were given ten minutes to select four pieces they may wish to add to a personal audio collection. It is important to note that preview cues in our prototype are not implemented as twenty second samples, as in the majority of current Web music sites. Preview cues start at the beginning of a piece and will play to the end of that piece unless the user makes another cue selection or stops the cue's playback. Before starting the trials, participants were provided with training for each of the interface functions. The training data set was pop rock music rather than classical. In this way, users were given exposure to the interface concepts without being exposed to the test data set.

In each of the studies, we looked at performance and perception of experience. Performance was measured by total time to complete the task, as well as by path exploration behaviours, such as the number of selections made, the depth of a path followed (advance clicks), and "back outs:" making a selection (selecting a node), but not following that path into that area of the domain (not selecting any of the leaves in that node). After each trial, we also ran a post task questionnaire to investigate users' responses to the interface.

In the between groups trials, we had 8 participants in each group. Both groups were gender and age balanced. We also balanced for "geek factor," balancing male and female computer science with non-computer science/computer-centric participants. In the within group evaluation, we ran another 8 participants. This group was also gender and age balanced. Exposure to the interfaces was counter-balanced. While we measured performance in this trial just as we did with the between groups, we were more interested in how the participants responded to each interface subjectively, given the experience/choice of one or multiple preview cues.

Between Groups

Quantitative Results. There was no significant difference in performance between multiple cues and single cues interfaces. As shown in the graphs of Figure 3 (a), only 9.7% more cues were triggered in the multiple cues interface (mean: 70.6, standard deviation: 3.18) compared to the single cue interface (mean: 77.5, standard deviation: 2.89). With three times the cues available, this figure would appear to indicate that their availability is unnecessary. This is supported by the results of a t-Test of difference, giving a high value of p = 0.659 (F = 1.21, T = 0.45, df = 13).

As the dataset was the same for both groups in the study, and only the number of available preview cues differed, it is reasonable to consider that the overall number of path selections would be similar and this is also shown in Figure 3 (b). There are actually marginally fewer clicks/selections within the multiple-cue interface (mean 22.3, standard deviation 8.6) than in the single cue interface (mean 22.9, standard deviation 6.7).

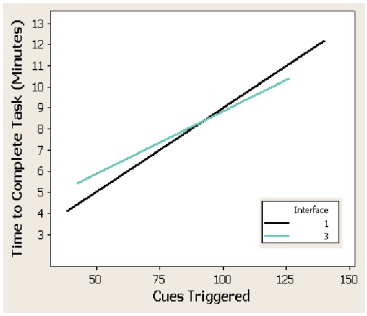
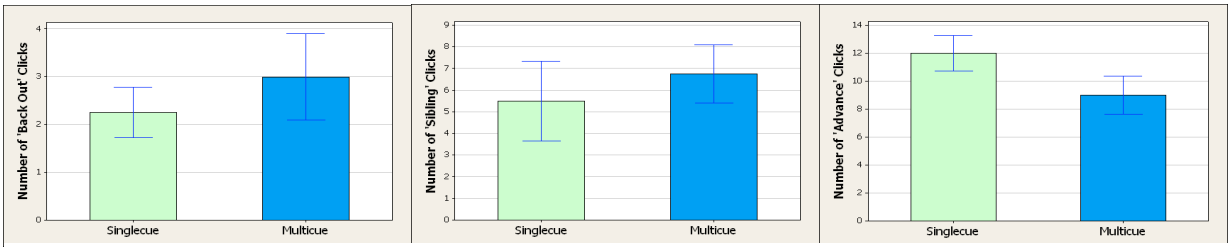


Figure 4. Correlation of Time spent on task and number of cues heard per interface.

The total time to complete the task is closely related to the number of cues fired as can be seen in Figure 4. Participants spent around an extra 12% of time on the multiple-cue interface (mean: 8 minutes, standard deviation: 1 minute 54 seconds) than on the single cue interface (mean: 6 minutes 41 seconds, standard deviation 2 minutes 47 seconds). The correlations also show, however, that having three samples available leads to a slower increase in time spent on task completion compared to cues triggered. This appears to imply that participants spent more time listening to a few samples, but spent less time on an individual sample when listening to a large range of samples.

The other measure of our multiple cue interface was backing out from a path choice, that is going back up the hierarchy after a selection has been made rather than



(a) Mean 'Back Outs' by interface (b) Mean 'Sibling' clicks by interface (c) Mean 'Advance' clicks by interface

Figure 5. Between groups performance comparisons. Error bars show ±1 standard error.

going deeper into that selection. As Figure 5 (a) shows the average number of back outs is larger and more varied for the multicue interface (mean: 3, standard deviation: 2.6) than the single cue interface (mean: 2.25, standard deviation 1). A similar trend is visible in the number of sibling clicks (Figure 5 (b)), defined by clicking on a number of items within the same list, such as composers. This behavior was also noted during the experiment when participants were given the multicue interface. Participants appeared to click on different items in a list, while listening to the larger range of samples.

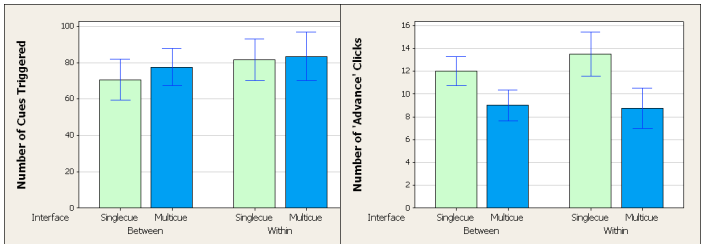
Advance clicks represent a selection followed by an exploration into the list of attributes that are expanded on the click/selection of a label. The comparison of average Advance clicks (Figure 5 (c)) between interfaces shows that users of the single preview cue interface explored down more paths than in the multicue interface. Participants performed Advance clicks 25% less when using the multiple-cue interface (mean: 9, standard deviation 3.9) than when using the single-cue interface (mean: 12, standard deviation 3.6).

Qualitative Measures. Users' responses to the interfaces also indicated that users were enjoying the interfaces and liked the new availability of audio information whether they had the single or the multicue interfaces. The sufficiency of samples was discussed with each participant and all of those who used the multicue interface indicated satisfaction, with only one member suggesting that the number be reduced. The single cue participants were split, 40% suggesting that the single audio cues were not necessarily sufficient for exploring the music. One participant, however, suggested that the cues be removed completely; this user was noted to have a high level of domain knowledge and expressed neither wanting nor needing the cues to explore the domain.

Within Group

Again, there was no significant difference in performance between the single-cue and the multiple-cue versions of the interface. This is shown in Figure 6. An analysis across both interfaces between the two groups for the number of cues triggered showed a low significance of p=0.46 (F = 0.74, T = -0.75, df = 29). The p-values resulted in a range between 0.32 and 0.78 and so any learning effect between interfaces is shown to be insignificant. The graphs show, however, that there is a small learning effect for these interfaces. Figure 6 (a) shows that the mean number of clicks increased after interacting with the first interface, whether the first interface was the single or multicue treatment. Similarly Figure 6 (b) shows the means of advanced clicks for the between and within group tests, and in both cases the number of advance clicks decreased when using the multicue interface. However, the average significance levels of these comparisons remains low. The most significant finding ($p < 0.1$) is the number of Advance clicks in the within group (F = 1.19, T = 1.79, df = 13).

In the within group, the user preference and the benefits of each interface were primarily discussed in the interview. Each participant's preferred interface was the interface that they experienced second. A majority of the participants suggested that having three icons gave a better representation of each category. Those that sited the single-cue as being their favorite interface said that it was a cleaner, or less 'cluttered' interface. Some of the participants who selected the single cue interface also reported that they found the multicue interface initially somewhat harder to use: they had, at the start of their trial, found it "tricky" to avoid accidentally triggering unwanted samples.



(a) Mean number of cues triggered (b) Mean number of 'Advance' click

Figure 6. Comparison of means of each interface between the study groups. Error bars show ±1 standard error.

General Observations

Although this effect was not formally measured, use of preview cues seemed to go down in both studies where a user's domain knowledge was greater. We noticed this in the training sets in particular, since only one of our participants was highly knowledgeable in the classical music domain and most of the participants had greater knowledge in the popular music domain, the source of our training set content. Both participants interaction and comments during the training set seemed to demonstrate that interest in exploring the samples appeared to be lower than in the classical set. Similarly, occasionally participants would indicate that the recognized a particular composer in the classical music set and they would select that domain area directly without first exploring the cues associated with it. In contrast to this reduction of preview cue use, as users become more familiar with the mechanism, the use of preview cues increases in larger (more general) regions of the domain. This was the case with the multicue treatment in particular.

DISCUSSION

Our results indicate that our hypotheses were shown to hold: the differences between the single cue and multicue interfaces for task performance are insignificant. This was the case in both the between and within group studies. Similarly, as we anticipated, in the multicue interface, participants explored the domain differently: they

investigated fewer paths as deeply in the multicue interface; more paths, more deeply in the single cue interface. This behaviour suggests that while there is no significant quantitative difference between treatments, there is a qualitative difference in patterns of exploration.

An additional pattern seen in the data is based on the number of brush actions observed in the multicue interface trials. As the one goes deeper into the hierarchy, and the number of possible cues available for each region of a domain area decreases, so too does the need for multiple cues decrease. Multiple cues provide greater benefits where there are more choices to make within lesser-known areas. This finding along with the decreased use of preview cues in cases where the user seems to be more familiar with the domain shows that use of multiple cues become more prominent when a person is in a relatively large, unfamiliar region of the domain. In other words, the value of multicues over single preview cues seems to be proportional to the area of the domain they represent and the level of user knowledge of the domain.

This pattern of behaviour is an interesting quasi confirmation of the design intent of preview cues as a tool to *help* users access areas of a domain where they are less familiar with the lexicon, organization or material of a domain. If they do not need the assistance, they do not need to avail themselves of it. Likewise, if designers wish to offer multiple cues, they may wish to consider two approaches: proportional representation of cues per area label or user-determined selection, on demand, of the number of cues they wish available. Proportional representation has the advantage of communicating additional information about the scope of that area of the domain. The number of cues associated with a label may either represent the state of the data available itself (one cue for every 20 pieces in that category) or the number of subcategories available for that part of the domain (3 cues for the 3 subcategories associated with that category once expanded). Even if only one preview cue is offered by default, such rules would be appropriate for determining how best to represent more cues on demand, especially since representing more than one icon per label persistently has screen estate costs associated with that display.

On the other hand, as we have seen, one cue, statistically, does just as well as several to help a user triage a domain space where there had previously been little or no access. Increased access was repeatedly commented upon by the majority of our participants: they were now able to explore a domain that had previously been effectively inaccessible. There is a considerable payback for designers interested in improving access to their content resources by deploying preview cues: people who previously had difficulty accessing a domain are now empowered to access it on their own, whether this content is part of a library or a music store.

FUTURE WORK

Preview cues are neither domain nor media specific. We are currently involved in a library project to associate preview cues in domain browsers of for both fine art collections and a film database. We plan longitudinal trials with the browsers in school libraries later this year.

While preview cues are not about teaching someone what the finer points are in being able to construct a concrete definition about one category vs. another, such as being able to articulate the differences between a concerto or a serenade by listening to preview cues of each, the trials will also give us insight into preview cues' role in improving domain knowledge. A person using our classical music browser may discover, for instance, that they seem to have a preference for a certain period and style of music. If that person can now walk into a record store and say "I'm interested in some recordings of baroque serenades – what have you got?" that person has gained a way to communicate some domain understanding from one context (the preview cue browser) to another (the store). Similarly they may also wish to know something more about the categories of their choices. A person who seems to choose many baroque concertos may want to know about those terms. In this respect, preview cues, and the choices they enable, may act as a mechanism for helping the user explore other attributes of the domain space if they wish; preview cues mean that they do *not* have to have that knowledge for access, but they can begin to build it with preview cues.

CONCLUSIONS

In this paper we have presented preview cues, a lightweight multimodal interaction technique for previewing the kind of information associated with a domain area. The preview cue thereby helps users determine whether or not an area of the domain is of interest for further exploration. Preview cues can be constructed from a range of multimedia content from still images to video. As a case study, we have shown that preview cues work well to enhance access to music by matching music audio with categories within the domain. In particular we have shown that, there is no significant difference in task performance between offering one or several cues per label, but that there are behavioural differences in exploration patterns.

Preview cues are a simple, lightweight technique, yet despite the power of the technique to improve domain access, we have seen little related work in the literature for cues that reflect an *area* rather than an *instance* in a domain. Despite the conceptual simplicity of the technique, there are several open questions to consider, such as the method for determining cues for selection: random, recommended or representational. Likewise, if multiple preview cues are desired, designers need to consider what kind of representation technique they wish

to use: fixed number, proportional, or dynamically, on demand.

Preview cues critically provide information to enable users with interest in but not great knowledge of a multimedia domain to be able to triage the domain rapidly to find content of interest. As our participants have reiterated, the presence of a preview cue has meant the difference between improved access to a domain or none at all. As such, preview cues offer a low-cost, multimodal means for improving domain access.

ACKNOWLEDGMENTS

We wish to thank all the participants in the user studies. Thanks as well to the folks of the IAM lab and Shengdong Zhao for their assistance. Thanks to Alan Dix, Tom Rodden and Andy Cockburn as well for their discussions with us about preview cues.

REFERENCES

1. Berners-Lee, T., Hendler, J. and Lasilia, O. The Semantic Web *Scientific American*, 2001. http://www.scientificamerican.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&catID=2.
2. Brewster, S.A. Using Nonspeech Sounds to Provide Navigation Cues. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *5* (3).1998, 224 - 259.
3. Chai, W. and Vercoe, B., Structural Analysis of Musical Signals for Indexing and Thumbnailing. In *Proceedings of the third ACM/IEEE-CS joint conference on Digital libraries*, Houstan, Texas, 2003: 27 - 34.
4. Cilibrasi, R. and Vitanyi, P. Clustering by Compression. *CWI*, cs.CV/0312044 (2003), 2003,
5. Crease, M. and Brewster, S.A., Making Progress with Sounds - the Design & Evaluation of an Audio Progress Bar. In *International Conference on Auditory Display '98*, University of Glasgow, UK, 1998.
6. Discogs. Database of Electronic Music Web Site: Http://Www.Discogs.Com/.
7. Eptionic. Epitonic.Com: Hi Quality Free and Legal Mp3 Music. Web Site: Http://Www.Epitonic.Com/.
8. Foote, J., T. Content-Based Retrieval of Music and Audio. in al., C.-C.J.K.e. ed. *Multimedia Storage and Archiving Systems Ii, Proc. Of Spie,* 1997, 138 - 147.
9. Herlocker, J.L., Konstan, J.A., Terveen, L.G. and Riedl, J.T. Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems (TOIS)*, *22* (1) 2004: 5 - 53.
10. IBM, T.J.W.R.C. The Glass Engine Project, Http://Www.Philipglass.Com/Glassengine/. 2002.
11. Li, T., Ogihara, M. and Li, Q., A Comparative Study on Content-Based Music Genre Classification. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, (Toronto, Canada, 2003), Annual ACM Conference on Research and Development in Information Retrieval.
12. Libbey, T. and Rostropovice, M. The Npr Guide to Building a Classical Cd Collection, Workman Publishing, New York, 1999.
13. Marshall, C.C. and Shipman, F.M., Spatial Hypertext and the Practice of Information Triage. In *Proceedings of the eighth ACM conference on Hypertext*, Southampton, United Kingdom, 1997: 124 - 133.
14. Oviatt, S., Cohen, P., Wu, L., Duncan, L., Suhm, B., Bers, J., Holzman, T., Winograd, T., Landay, J., Larson, J. and Ferro, D. Designing the User Interface for Multimodal Speech and Pen-Based Gesture Applications: State-of-the-Art Systems and Future Research Directions. *Human-Computer Interaction*, *15* (4).2000: 263 - 322.
15. Pollitt, A.S., Ellis, G.P. and Smith, M.P. Hibrowse for Bibliographic Databases. *Journal of Information Science*, *20* (6).1994: 413 - 426.
16. Robertson, G.G., Mackinlay , J.D. and Card, S.K. Cone Trees: animated 3d Visualizations of Hierarchical Information *Proceedings of the SIGCHI conference on Human factors in computing systems: Reaching through technology*, New Orleans, Louisiana, United States, 1991: 189 - 194.
17. Schmandt, C., Audio Hallway: A Virtual Acoustic Environment for Browsing. In *Proceedings of the 11th annual ACM symposium on User interface software and technology*, San Francisco, California, United States, 1998:163 - 170.
18. schraefel, m.c., Karam, M. and Zhao, S., Listen to the Music: Audio Preview Cues for Exploration of Online Music. In *Proceedings of Interact 2003 - Bringing the Bits Together,* Zurich, Switzerland, 2003: 4pp.
19. schraefel, m.c., Karam, M. and Zhao, S. Mspace: Interaction Design for User-Determined, Adaptable Domain Exploration in Hypermedia. De Bra, P. ed. *Workshop on Adaptive Hypermedia and Adaptive Web Based Systems*, Nottingham, UK, 2003, 217 - 235.
20. schraefel, m.c., Shadbolt, N.R., Gibbins, N., Glaser, H. and Harris, S. Cs Aktive Space: Representing Computer Science in the Semantic Web. *Proceedings of World Wide Web Conference 2004*, New York, New York, USA, 2004.
21. Terry, M. and Mynatt, E.D., Side Views: Persistent, on-Demand Previews for Open-Ended Tasks. In *Proc. of UIST*, 2002: 71 - 80.
22. Tweedie, L., Spence, B., Williams, D. and Bhogal, R., The Attribute Explorer. In *Conference companion on Human factors in computing systems*. Boston, Massachusetts, United States, 1994: 435 - 436.
23. University, I. Indiana University Music Library Variations: Http://Www.Dlib.Indiana.Edu/Variations/, Indiana.
24. Wold, E., Blum, T., Keislar, D. and Wheaton, J. Content-Based Classification, Search, and Retrieval of Audio. *IEEE MultiMedia*, *3* (3) 1996: 27 - 36.