Visualizing Amazon.com’s Purchase Circles

David DeMumbrum, Jeannie Lee, William Stark, Bessie Yang
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332 USA
{ddemumbr, jeannie, wstark, bessie}@cc.gatech.edu

ABSTRACT
Amazon is a well-known online company that sells products such as books and music. It also tracks the purchasing patterns of a variety of groups including private corporations, government organizations, and geographic areas. Amazon defines each of these groups as a “purchase circle.” For each purchase circle, Amazon lists the best-selling items in the Books, Music, Video, DVDs, and Electronics product categories. Our objective is to create a dynamic visualization of Amazon’s purchase circles that focuses on looking at the Top 10 music titles and genres that are popular in selected U.S. cities. We present a visualization known as CityPrints, a dynamic query-based tool for producing color-coded visual representations of purchase circles data. CityPrints allows users to quickly compare popular titles in different U.S. cities, identify which music genres are popular in a given city, and rank cities according to how popular a given music genre is in that city.

Keywords
information visualization, purchasing patterns, visualization, dynamic queries

INTRODUCTION
Amazon is a well-known online company (http://www.amazon.com) that sells a wide variety of products such as books, movies, and music online. One of its more interesting characteristics is a purchase circles feature which tracks the purchasing patterns of a variety of groups including private corporations, non-profit and professional organizations, colleges and universities, government organizations, and geographic locations. Amazon defines each of these groups as a “purchase circle.”

For each purchase circle, Amazon lists the best-selling items in the Books, Music, Video, DVDs, and Electronics product categories, but it does not rely solely on overall sales when compiling its lists. Instead, Amazon creates its purchase circle lists by aggregating data on items that have been shipped to a particular location or ordered from a particular domain name. It then modifies this information to create lists of items that are more popular for that group in comparison with the general population. The update frequency of purchase circles varies from weekly to monthly, depending on the size of the purchase circle [1]. Currently, users who are interested in looking at a specific purchase circle must go to Amazon’s Web site and navigate to the desired purchase circle and product type. This results in a list containing information about the Top 10 items that are popular for that purchase circle. For example, in music items, the Top 10 list contains information about the title of a music album, the artist’s name, pricing, availability, and average customer rating. With this current system, users have no way of performing side-by-side comparisons of multiple purchase circles or quickly identifying what types of music are popular in a given purchase circle.

Our dynamic visualization tool, known as CityPrints, is designed to provide these features using an interface that is simple and straightforward. For demonstration purposes, our initial version of the CityPrints tool is focused on the category of music and depicts purchase circle data for a limited number of U.S. cities. Through a list box and an assortment of radio buttons, the user selects a set of cities (numbering between one and five) and a sorting method. In response, the CityPrints tool displays for each selected city a vertical stack of ten horizontal colored bars, one for each of the top ten most popular music selections for that city. Each color corresponds to one of ten unique musical genres. Clicking on any one individual color bar results in the details of the represented selection being displayed in a text window to the left of the set of CityPrints, including the selection’s title, artist, genre, year of release, and sales rank.

Our target users are people who are interested in the following kinds of tasks:

- Identify the most popular music title(s) in a specific U.S. city
- Given a specific music title/genre, identify the cities in which it is popular
- Rank the popularity of music genres for selected U.S. cities
• Compare popular titles/genres between multiple U.S. cities

In the sections that follow we will describe our design decisions, related work that contributed to the development of CityPrints, an overview of the CityPrints visualization tool, and finally aspects we would like to focus on during future development of CityPrints.

**DESIGN AND IMPLEMENTATION**

Our group hypothesized that Amazon’s purchase circles data would confirm common presumptions about the popularity of musical genres within certain cities. For example, we expected that Nashville, TN, would feature more country music titles in its Top 10 list than most other U.S. cities. We also hypothesized that some cities would share matched titles in their Top 10 lists. We originally hoped to look more closely at such matches in order to determine if cities containing matched items held additional musical preferences in common. However, as we gathered data from the Amazon Web site, we realized that because very few cities shared matched music titles in their Top 10 lists, there was little opportunity to explore overlapping music tastes by looking only at matched titles. We consequently decided to focus on visualizing comparisons between cities based on musical genre, while providing details on demand for specific albums. To keep our visualization simple, we limited the number of genres to the following ten: Blues/Funk, Classical, Country/Folk, Dance/Electronica, Jazz, Miscellaneous, Rap/R&B, Rock/Pop, Soundtracks, and World.

**Displaying Cities**

For our dynamic visualization, we wanted to enable our users to compare the music preferences of different cities using as little cognitive effort as possible. Therefore, we chose to capitalize on the human perceptual ability to process certain attributes such as color pre-attentively [7].

In CityPrints, each city is represented by its own CityPrint, a graphical column consisting of 10 colored rows. Each of the rows corresponds to a music title in the city’s Top 10 list and is color-coded according to each title’s associated genre. For example, the CityPrint in Figure 1 for Chicago contains six red rows as well as one orange, green, pink, and blue row for a total of ten rows. Red rows represent Rock titles, orange represents a Dance/Electronica title, green represents a Miscellaneous title, pink represents a Classical title, and blue represents a Jazz title. This method of encoding provides a simple and quick way for users to see how genres are distributed throughout a city’s Top 10 list as well as compare multiple cities to see if they exhibit preferences for particular musical genres.

One drawback to this technique pertains to its heavy reliance on color as an information-encoding mechanism. In order to make proper use of CityPrints, users must differentiate amongst different colors in the display. CityPrints would therefore be ill suited for use by colorblind individuals and/or users working on a monochrome display. A simple resolution to this problem might involve adding text labels to CityPrint rows in order to specify the genre represented by each row. Users forced to rely upon such labels would miss the pre-attentive processing advantages afforded by color encoding, but they would nevertheless have access to all information within the display. Another potential issue is the fact that while users can quickly discern how many genres are represented in a particular city, they might not be able to identify those specific genres without referring to a legend first. However, we anticipate that this would not be a serious issue, especially for experienced users. A simple feature that could help alleviate this problem would also be to allow users to customize the colors for each music genre according to their own preferences.

**User Queries**

In order to provide users with continuous updates to specified queries, we based the interactive properties of CityPrints on the dynamic queries pioneered by Shneiderman [6]. However, the limited-case nominal structure of data visualized by CityPrints suggested an alternative to the slider control mechanism classically associated with dynamic query applications. CityPrints instead provides users with a quick means of obtaining custom visualizations based on their use of buttons and lists to select the names of cities and musical genres they wish to see represented. For example, if a user wants to compare music preferences in New York City, NY, and Seattle, WA, she can select those cities within a list, and their corresponding CityPrints will appear on the display area of the visualization, shown in Figure 2.
Alternatively, a user interested in determining which cities most strongly prefer a particular type of music can pick a genre (e.g., “Jazz”) from the appropriate list and obtain a collection of CityPrints corresponding to the five cities in which the selected genre is most popular. CityPrints displayed as the “Top 5 most popular” for this purpose would be automatically selected by an application-based algorithm that performs calculations to determine which five cities contain the largest overall number of Jazz albums in the CityPrints data set. Only cities with at least one jazz album can be displayed, and if there are fewer than five cities that meet this criterion then only those cities are displayed.

We also considered implementing a variation on this scheme that would choose a Top 5 collection based on what we call a popularity index. In this scheme, the Top 5 CityPrints chosen for display would be based not only on the total number of albums in a particular genre featured within a city’s Top 10 list, but also on the overall rank number assigned to each album (a figure Amazon computes based on sales volume). Complexities associated with implementing this idea, combined with a minimal impact on the general interest value of our data set, caused us to bypass this idea for the time being.

While the present CityPrints interface does not quite fit the true definition of dynamic queries due to the need for a user to press a button in order to refresh the display, future versions of our prototype would allow users to see their query results immediately.

**Displaying Details**

One challenge present throughout the development of CityPrints concerned its ability to deliver detailed information (for example, album title, artist, and so forth) about all of the music titles represented in the display. We considered using a focus + context method, similar to the Table Lens [5], of expanding a selected row (representing a music title) to show the details. Deselecting a row would cause it to return to a normal-sized colored bar, and multiple rows could be selected at a time. This would avoid requiring users to manage multiple windowed displays. However, the disorienting liabilities associated with this focus + context method could potentially inhibit the user’s ability to formulate comparisons of music titles by rank between various cities.

We eventually chose an overview and detail approach where users may obtain “details on demand” for an album by clicking on any row within a particular CityPrint. To facilitate construction of a prototype, we placed the mouse-over details into a tiled window that appears in the upper-left corner of the CityPrints application. Thus, within the current prototype, selected CityPrints comprise the main display, while details associated with individual albums appear inside smaller fixed window off to the side, similar to the tiled multilevel browser described in [4]. This provides a simple solution to the detail problem, but at the same time adds drawbacks of taking up valuable screen space, as well as dividing a detail-seeking user’s attention between two separate displays. Another disadvantage when compared to the focus + context method described previously is that the details for only one music title can be displayed at a time.

![Los Angeles, CA](image)

| Title: EP 2 |
| Artist: Zero 7 |
| Genre: Dance/Electronic |
| Year Released: 2001 |
| Sales Rank: #1 |

**Figure 3.** Details for the Los Angeles purchase circle’s top music title

**Sorting**

The default sorting of each CityPrint follows the Amazon-derived ranking of a city’s Top 10 music titles by sales volume. Thus, in any CityPrint, the number one selling album on a city’s Top 10 list appears by default in the topmost row, while the number 10 selling album appears in the bottom row. One problem is that users might find it
difficult to gather meaningful information using this approach, especially if they are not interested in the relative rankings of music titles within a particular city. For example, if a user wants to determine which city has the most Rock/Pop albums in its Top 10 list, he may have to resort to manually counting each of the red-colored bars which are scattered across each of the CityPrints. To address this, CityPrints’ sorting features allow users to modify the ordering of rows within each CityPrint. Users can choose to sort each CityPrint by the default sales ranking, user-specified genre, and overall frequency of each music genre.

Users most interested in particular types of music can utilize the sort feature to modify the ordering of titles according to a specific genre. In this way, the sort feature allows users to automatically cluster together albums within particular genres of interest. For example, if a user chooses to sort by the Country/Folk music genre, then all albums within the Country/Folk genre collect at the top of all visible CityPrints, creating something of an inverted bar chart in Figure 4. In addition to clustering, this technique also allows users to explore particular genres of interest within the context of a genre-based display. For example, the music titles that do not fall under the Country/Folk genre are also clustered together by their corresponding genre, and the genre clusters are ordered alphabetically from top to bottom by the name of the genre (so Classical would appear below Country/Folk which is the selected music genre, but above Dance/Electronica).

In the example shown in Figure 4, users can see that Nashville has the most Country/Folk albums, but they can also quickly see that Atlanta has the most Rock/Pop albums, followed by Dallas, while Nashville has no Rock/Pop albums at all in its Top 10 list. Note that this type of “dual” bar chart occurred because none of the three cities had Soundtracks or World music in their Top 10 lists. However, even when this is not the case, users can still quickly compare the relative widths of each colored genre cluster to determine its popularity for each city without necessarily having to sort by each genre.

Users can also choose to sort selected CityPrints by the overall frequency of each music genre in their respective Top 10 lists. Each CityPrint is sorted with the highest frequency genre clustered at the top, followed by the genre with the next highest number of music titles in the Top 10 list, and so on. This provides users with an alternative method of determining which music genres are most popular for particular cities. For example, in Figure 5, users can quickly see that Rock/Pop is the most popular music genre in Boston with five music titles, followed by

![Figure 4. Selected CityPrints sorted by the Country/Folk genre](image4.png)

Classical and Dance/Electronica with two music titles each. In the case of a tie, the genres are placed in alphabetical order from top to bottom.

Unlike the Sort by Genre option, which aids the task of identifying how popular a specific music genre is for selected cities, the Sort by Frequency option aids the task of ranking the popularity of all of the music genres for the selected cities.

By using the sorting features, users can view the relative popularity of a chosen genre across multiple cities while maintaining the ability to view other genres popular in all cities on display. This approach is similar to the work of

![Figure 5. Selected CityPrints sorted by their most popular music genres](image5.png)
Keim et al. [3], whose pixel bar charts also relied heavily upon color as a means of encoding data and used equal-height bars in order to use screen space efficiently. CityPrints differs from pixel bar charts in that it uses graphical rows rather than pixels to represent data items. Since it currently involves only nominal data, CityPrints also does not use the pixel bar chart’s equal-height/varied-width mechanism to represent quantitative data.

**Implementation**

CityPrints was implemented in Visual Basic under various versions of the Windows operating system and currently runs on most versions of Windows. This approach was taken to minimize development time and arrive at a workable prototype rapidly.

The implementation currently has no database backend. All CityPrints and selection details were hard coded for the purpose of demonstrating application capabilities.

For future prototypes, we intend to port the implementation to Java to ensure that the program is platform independent and further employ the use of 2D graphics libraries to reduce image loading and processing overhead during program execution.

**FUTURE DIRECTIONS**

Although our current implementation of CityPrints does meet the general objective of expressing a city’s music preferences in a simple and straightforward manner, we have identified a number of potential improvements.

First and foremost, we would like future versions of CityPrints to include a completed implementation of a mouse-over feature for displaying music title details. Such an improvement would resolve some of the more irksome problems we encountered using overview and detail, in the sense that users could seamlessly obtain information associated with individual music titles without having to consult an extra display off to the side. We envision such a feature being similar to a “gloss” in the Fluid Links system [8], in that it would show additional information without consuming screen space or occluding the primary CityPrints display.

Scalability is another issue we plan to address in future versions which applies to both displaying data and entering queries. In its current form, CityPrints allows users to display up to 5 cities at a time, with each city arranged in alphabetical order from left to right. We would like to allow users the ability to select and view more cities simultaneously, as well as enable them to dynamically change the position of a CityPrint instead of locking it into a predetermined slot. Users could then cluster CityPrints in a flexible manner without being forced into a grid-based, alphabetically arranged format. The size of each CityPrint could also be dynamically changed to accommodate the number of displayed cities, although this would require careful consideration of how large each CityPrint must be to be useful.

In terms of the scalability of entering user queries, users currently select cities using a dropdown box which, while acceptable for our current prototype that includes only 10 cities, would complicate selection tremendously if we significantly increased the number of available locations. One possible solution might involve using a map to display a larger number of cities. This approach would provide users with a natural and intuitive means of city selection (provided they were familiar with the geography of the United States), and also contribute the added variable of geographic location to possibly enrich the overall display of purchase circle information. For example, with a map, users could better consider the possible impact of geographic location on a city’s musical preferences. A map-based representation would complement our slated future implementation of dynamic query mechanisms. For example, it could allow users to request a display of CityPrints that contain at least two albums belonging to the Blues/Funk genre.

Our consideration of such ideas was influenced by the work of Dang, North, and Shneiderman [2], whose Dynamaps made use of dynamic queries through brushing techniques applied to choropleth maps. Our design would take a similar approach in that users can specify a query that causes the resulting cities to light up in the map display and the corresponding CityPrints to appear in an adjacent display. A more advanced version of this might allow users to select cities by highlighting a rectangular region of the map (for example, selecting only cities in the Southeast United States) with the mouse cursor and then entering queries that apply only to the selected cities. CityPrints would differ from Dynamaps in that we would concentrate only on the display of cities for selection purposes rather than using a choropleth map.

A drawback associated with using a map selection scheme concerns the nontrivial difficulty of differentiating amongst (and potentially labeling) cities in close geographic proximity to one another (for example, Chicago, IL, and Gary, IN). A poorly implemented map-based selection mechanism could require the user to execute some extremely tricky mouse movements! A zooming interface could ease this problem somewhat; however, we would carefully need to weigh the positive and negative aspects of such an elaborate user interface work in order to avoid confusing users and cluttering the CityPrints application environment.
In addition to an enhanced city selection mechanism, we would like to further enhance the existing CityPrints prototype to display time series information associated with each city. We found interesting the prospect of allowing users to view a city’s current CityPrint and compare it with CityPrints of the past (for example, 6 months ago, 1 year ago, 2 years ago, and so on). Such a feature would allow users to view how location-based music preferences may have shifted over time.

CONCLUSIONS
We have designed and implemented a dynamic visualization tool named CityPrints that allows users to view and compare music preferences in selected U.S. cities, as reported in Amazon’s purchase circles. Our objective was to organize and display a city’s music preferences using a simple and straightforward interface. We feel that we have achieved this objective with our current implementation, which allows users to select cities by name and popular music genres as well as sort each CityPrint according to sales ranking, user-specified genre, and overall frequency of each genre in a CityPrint. However, there are still a number of potential improvements and added features that can improve the functionality and usefulness of CityPrints. These include the use of semi-transparent windows to display information about selected music titles, enhanced scalability, geographic representation of cities, selection of cities in a manner more closely in-line with the dynamic queries paradigm, and the incorporation of time series data.

REFERENCES