mashpoint: Supporting Data-centric Navigation on the Web

Igor Popov
School of Electronics and
Computer Science
University of Southampton
SO17 1BJ, Southampton, UK
ip2g09@ecs.soton.ac.uk

CHI'12, May 5–10, 2012, Austin, Texas, USA.

Abstract
Large numbers of Web sites support rich data-centric features to explore and interact with data en masse. For example, online shopping or travel sites routinely offer users ways to filter, explore and interact with their data. The ability to find related information on other Web sites about selected subsets of the data, however, is currently compromised by the Web's inherent, one document at a time, navigation. In this paper we present work-in-progress on mashpoint, a framework that allows distributed data-powered Web applications to exchange subsets of their data, in effect enabling many-to-many navigation on the Web, at a more granular, data level. We hypothesise that allowing such navigation unlocks novel possibilities for information exploration and interaction on the Web. We present an initial prototype and discuss the opportunities and challenges of facilitating this kind of interaction.

Author Keywords
Web Navigation, Data Interaction, Information Exploration

ACM Classification Keywords
H.5.m [Information interfaces and presentation (e.g., HCI)]: Miscellaneous. See:
http://www.acm.org/about/class/1998/

Introduction

Many interactions available on the Web are inherently data-centric i.e they let us deal with multiple data simultaneously. Whenever we access our social networking sites, or browse for products on online shopping sites we are routinely provided with means to explore this data - we can filter our friend feeds, and browse through multiple products by selecting specific facets. While such data exploration tools allow us to explore the data provided in a particular service, our ability to quickly find related data on the Web is compromised, because the Web does not support navigation with multiple, granular pieces of information at once; rather it provides navigation from one single document to another.

To illustrate the problems of this limitation we consider the following example. Let us suppose we have two distinct online shopping sites we regularly consult when we are considering purchasing a new mobile phone: PhoneFind.com and ReviewPhone.net. PhoneFind offers us various data about mobile phones including data about type of phone, vendor, model and price. The web site offers us to navigate through its vast data about phones by either keyword search or filters applied on facets. For example, we can easily find phones from a particular vendor, and are at a maximum price of 200£with a monthly plan. Battery life is also important to us since we plan to use battery-heavy features, however we are interested more in customer reviews about the subject rather than official statistics. Unfortunately we notice that data about battery life is absent from PhoneFind; moreover the website offers no reviews from customers. ReviewPhone on the other hand does not provide a lot of technical specs about phones, however, it provides excellent reviews that includes statistics on battery life collected from real users. However, we cannot easily take or navigate with our selection of phones from PhoneFind and find the corresponding data on ReviewPhone; rather we have to do the hard and tedious work of copying the name of every phone we found on PhoneFind and search for the exact same phone on ReviewPhone. This process can be time consuming as well as error prone.

In this paper we propose a framework that extends browsing on the Web to facilitate many-to-many navigation on the Web. Many-to-many navigation has been proposed in number of data exploration tools, however our paper if the first to propose investigating such interactions in a distributed publishing space such is the Web. The paper is structured as follows. First we briefly describe related work. Afterward we present mashpoint[1], a prototype implementation of a framework that allows many-to-many navigation on the Web. The subsequent section briefly discusses implementation details. This if followed by a section examining the interaction challenges with the proposed approach. Finally, in the last section we conclude and discuss future work.

Related Work

In this paper we propose extending the one-to-one interaction model of browsing Web pages to support navigation with multiple data items from one page to another page. This implies that we no longer are constrained by navigating though documents, but rather we facilitating navigation between more granular sources of information or data. To enable such interactions, data must be first-class citizens on the Web, that is to be uniquely identifiable similarly to how every Web page is an identifiable resource on the current Web. The concepts of unique identification and data independent of document presentation are one of the key proposals behind having a

[1] http://mashpoint.net/

Semantic Web [1] and a Web of Linked Data[2]. From a interaction perspective, the many-to-many interaction we propose for sharing data between applications, while uncommon and novel on the Web, has conceptually been implemented elsewhere to support richer interaction with data. For example, faceted browsing [6] has been extended with pivoting (for example FacetLens [5]) which is used to re-focus the set of result items allowing the faceted browser to switch the focus of exploration. Set-oriented approaches such as the Parallax browser [3] allows many-to-many navigation of data on Freebase[2] an open wikipedia-like repository of structured data. Earlier work on visual query systems builds on a metaphor of parallel navigation. All of afore mentioned implementations are restricted to using this method of interaction within the confines of a single application; the work here, however, explores how such interaction can be embodied in a distributed model such as the Web.

mashpoint - Pivoting with Data between Web Pages: An Example

To better illustrate the type of interactions we want to support with mashpoint let us examine a simple example of two applications linked up using the mashpoint framework. Figure 1 depicts two distinct data-centric applications that are connected through mashpoint. The first application (Figure 1a) is an application showing levels of income per capita for countries in the world. In allows us to view population and income levels for countries, as well as to filter the countries by a number of facets: geographic area, income level, membership in international organisations etc. The other application is a similar application that allows us explore the birth rate vs. death rate in various countries. It allows us to filter the countries by geographical location and population size. Let us suppose we are exploring the income levels of different countries in the first application and we filter them by income level to get only countries that are classified as low income. We now are interested if there is potential correlation between income levels and birth rate vs. death rate in those low income countries. The first application however does not provide this particular data. An application adapted to the mashpoint framework, however, allows finding other applications that have data about the things that are in current focus (Figure 1b). Since they inherently talk about the same types of resources (in this case countries) the second application can "take" the selected countries from the first application and focus on them in the second application. Thus by pivoting with this data on the second application we can see the countries filtered in the first application in a scatter plot showing birth rate vs. death rate and conclude that while death rate in low income countries is fairly similar there is great variance in the birth rates.

Implementation

In this section I describe the implementation details of the prototype mashpoint framework. Figure 2 depicts the general architecture and approach. The framework allow a publisher of a data-powered Web site to subscribe their data to a discovery service, which is a repository of unique identifiers (URIs) of the real world objects[3]. In the process of registration, individual data items identified in an application need to be aligned to the identifiers in the portal. Based on the fact that they operate over same identifiers the portal then can recommend applications that can represent various data given any subset of URIs. To enable selections of the data to be shifted between

[2] http://www.freebase.com/

[3] Currently we are using Freebase identifiers to identify such real world objects

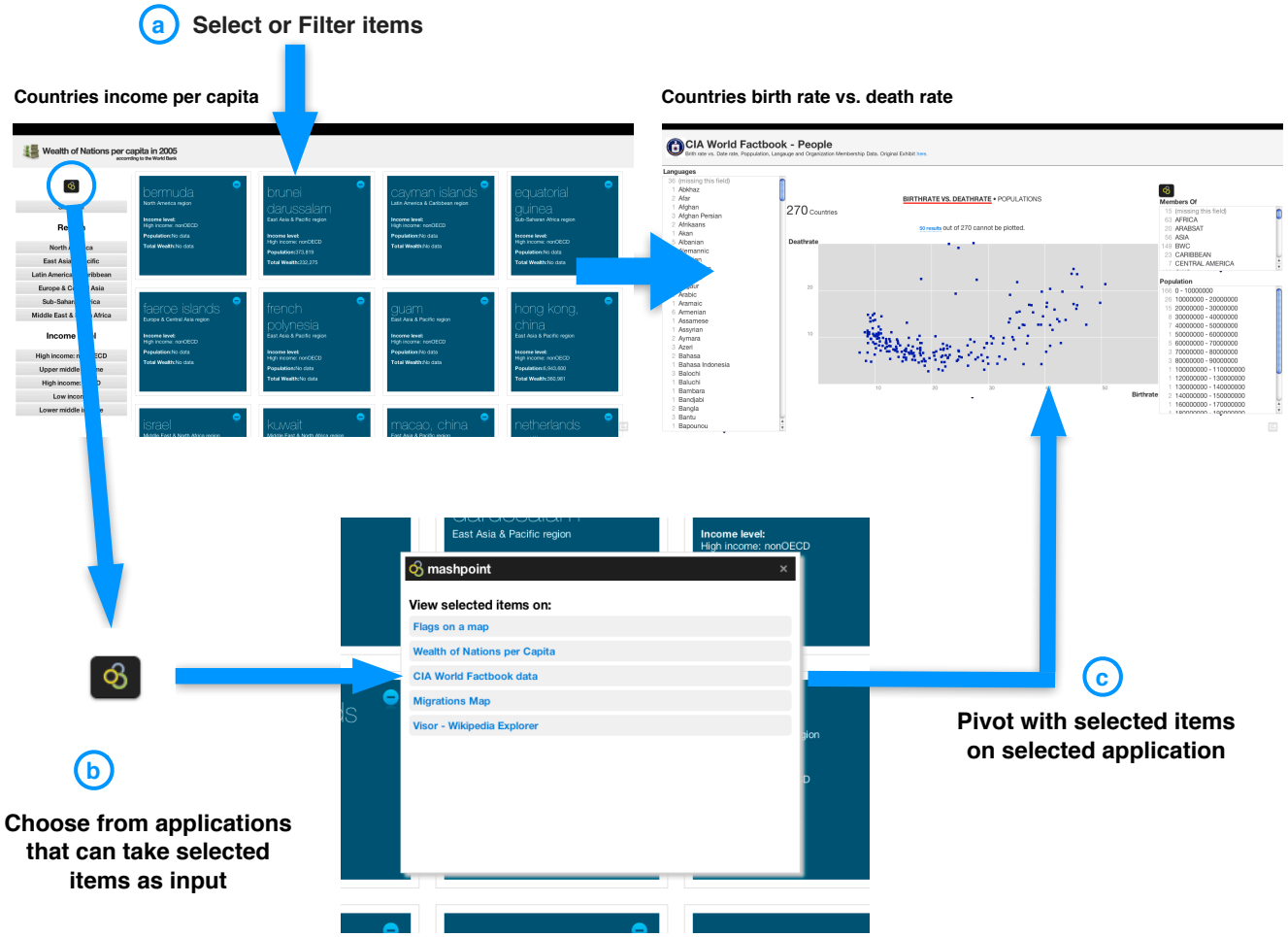

Figure 1: n example of pivoting with data between two applications in mashpoint.

applications, each application needs to be observe the following protocol:

- **Be able to select multiple resources.** An application in the framework should enable arbitrary selections of their data in order to navigate with that selection of data. Selection can be made either though facets or arbitrary selection and is left to the application publisher. This selection of items will then be passed on as input to another application.

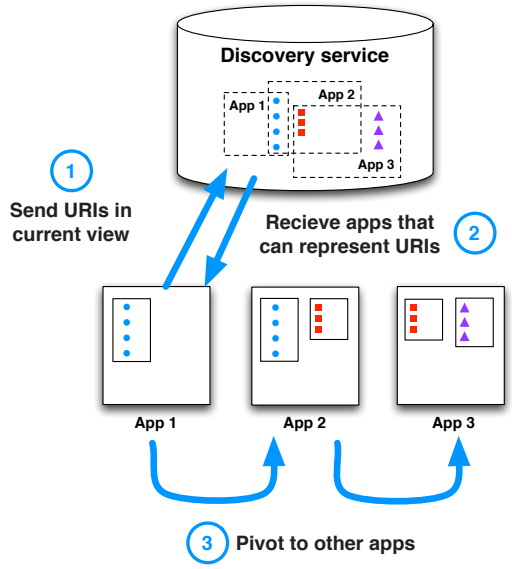

Figure 2: Architecture of the mashpoint framework.

- **Be able to represent multiple resources on input.** Application should be able to take any identifiers of the items over which they are built and be able to show a representation that corresponds to only those items. Another way to say this is that the application should be able to represent an arbitrary selection of its data on demand.

In mashpoint the currently selected items represent the state of the application. The state of the application is described in the URL of the application which holds a parameter containing information regarding which items are currently selected.

Discovery Service
The mashpoint discovery service is a repository of URI identifiers and information about registered applications. Applications can "subscribe" to the identifiers such that registration means an application can represent and show data about any subset of the identifiers it is subscribed to. Figure 2 symbolically depicts different collections of identifiers as dots, triangles and squares. For example App 1 is registered with the dot identifiers, as is App2. App2, additionally can take any subset of the square identifiers. Since mashpoint applications need to operate over the same URIs in order to pivot between each other, registration implies aligning their data identifiers to identifiers in the discovery service. The problem of aligning the data with a provider of identifiers is known as reconciliation of data, an services such as Google Refine[4] provide tools to reconcile data from different source. In the prototype of mashpoint we use Freebase identifiers and use Google Refine to reconcile data from new applications.

Navigating with data across applications
In order to enable navigating with data across applications, applications need to communicate and request information based on the current state of the application. To get other pivoting applications each application incorporates a small JavaScript widget that is

[4] http://code.google.com/p/google-refine/

able to parse the URL for identifiers and send them to the mashpoint portal (Figure 2-1) The mashpoint portal then sends a response of applications that can set their current states with those identifiers (Figure 2-2). The user is then presented with links to those applications with the identifiers in the URL parameter and can proceed to pivot to the selected application (Figure 2-3).

Interaction Challenges

Evaluating a Distributed Design

Unlike a other tools where data is viewed, browsed and manipulated within the context of a single application, browsing data with mashpoint is an entirely novel approach to data browsing where views of the data are provided by distributed applications that can be contributed by many publishers. Thus, evaluating how well users can combine and solve complex data queries using different applications is one of the research challenges we aim to explore with mashpoint.

Showing Context Between Applications

While mashpoint currently support only refocusing on semantically same resources (e.g. Countries) we plan to extend navigation that allows refocusing from related data, for example the facets from one application, to an application which has the facets as its main data item collection. To support such interaction, however, requires that we relate context information i.e. how the items in the current view relate to the items of the application from which we navigated.

Conclusion

In this paper we presented initial work on mashpoint - a framework that aims at supporting data interaction on the Web. Our initial prototype shows that such interactions are viable, an future work includes evaluating the usability and usefulness of the approach.

Acknowledgements

This work was supported by the EnAKTing project, funded by EPSRC project number EI/G008493/1.

References

[1] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web: Scientific american. *Scientific American*, May 2001.

[2] C. Bizer, T. Heath, and T. Berners-Lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 2009.

[3] D. F. Huynh and D. R. Karger. Parallax and companion: Set-based browsing for the data web. In *WWW Conference*. ACM, 2009.

[4] D. F. Huynh, D. R. Karger, and R. C. Miller. Exhibit: lightweight structured data publishing. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 737–746, New York, NY, USA, 2007. ACM.

[5] B. Lee, G. Smith, G. G. Robertson, M. Czerwinski, and D. S. Tan. Facetlens: exposing trends and relationships to support sensemaking within faceted datasets. In *Proceedings of the 27th international conference on Human factors in computing systems*, CHI '09, pages 1293–1302, New York, NY, USA, 2009. ACM.

[6] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '03, pages 401–408, New York, NY, USA, 2003. ACM.