Mashups: Patterns and Development Tools

Mathindri Pathiraja
Macquarie University
Sydney, Australia
Student no: 41364465
pathiraja.pathiraja@students.mq.edu.au

Abstract

In recent years major web services have opened their systems to outside use through the implementation of public APIs. As a result, web developers have begun to experiment with *Mashups* — software applications that merge separate APIs and data sources into one integrated interface. Because the APIs and data sources are publicly available, in principle anyone can create a Mashup. However, because relatively advanced programming languages are required to integrate these APIs, creating a Mashup still requires considerable programming expertise. I conducted a qualitative survey of high quality Mashups, as nominated in two popular directories, and examined how the Mashups made use of existing websites or improved upon them, how data from multiple websites were combined, and what kinds of user tasks these Mashups might be suitable for. I have described a set of patterns what I have found in our sample Mashups. This paper describes preliminary work in the uncovering of Mashup patterns in order to find new directions for the design of Mashup tools.

Keywords: Web2.0, Mashup, API, RSS, Patterns.

1 Introduction

Mashups are inspiring and interesting because they are an expression of how users make use of existing system in new and innovative ways. As defined on Wikipedia[1] *"Mashup is a Web application that combines data or functionality from two or more sources into a single integrated application. The term Mashup implies easy, fast integration, frequently done by access to open APIs and data sources to produce results that were not the original reason for producing the raw source data. An example of a Mashup is the use of cartographic data from Google Maps to add location information to real estate data, thereby creating a new and distinct Web service that was not originally provided by either source."* One of the earliest mashups was Paul Rademacher's HousingMaps.com, a combination of apartment rentals and houses for sale extracted from the online classifieds site craigslist.org, and displayed on Google Maps. On this site, housing ads are enriched through geo-referencing, both in terms of pragmatic utility (helping users find accommodations in a given area more easily) but also in terms of the demographic insight which emerges from the large dataset on craigslist that documents a spatially significant human activity, namely the processes involved in the acquisition of housing. HousingMaps was simple to create and instantly filled a need not provided by the original data source craigslist, exemplifying a new paradigm called Web 2.0 by Tim O'Reilly, a leading publisher of technical books and a trendspotter in the computer industry. For O'Reilly, Web 2.0 refers to new interoperable, interlocking type of services, where web sites would provide components (either data itself or data processing services) rather than finite, one-stop experiences. Users in this paradigm would, therefore, be free to combine online services in any way they choose. "Google Maps with craigslist is the first Web 2.0 application," O'Reilly explained at his 10 geospatial Where 2.0 conference in 2005 [7].

Adrian Holovaty's Chicago Crime[2] was, along with HousingMaps, perhaps the best known of the early Mashups. The site displays crime reports from a feed available on the Chicago Police Department's web page and geo-coded the reports onto a Google Map. Later, the site moved beyond simply identifying crime locations with colored pushpins and began linking the locations with stories from the Chicago Journal's police

[1] http://en.wikipedia.org/wiki/Web_mashups (22/05/09)

[2] *On January 31, 2008, Holovaty shut down the original Chicago Crime mashup, redirecting it to EveryBlock, a new project with expanded functionality and coverage of more cities.* http://www.everyblock.com

blotter, widening representations of space beyond plotted points to include narrative.
The need for a hacker's skill set also meant that creating these Mashups was an elite venture, only possible for those with programming knowledge. In fact, these early Mashups were also called "map hacks" or "mapping hacks,"[3] terms that were already in use in 2004 (the year before Housing Maps and Chicago Crime were released) by programmers Schuyler Erle, Rich Gibson and Jo Walsh in the website[4] for their book *Mapping Hacks* (subsequently published in 2005) [1]. While *Mapping Hacks* doesn't mention the word "mashups," a subsequent book by Gibson and Erle (2006) [3] clearly defines Mashups as a subset of mapping hacks, involving, as in music, combining two existing data sources, such as a news feed and a web map. The remainder of the book is devoted to mapping one's own data, or describing advanced techniques for dealing with Google Maps as a user, none of which are described as Mashups. This continues with the strict definition of Mashup: it must combine two or more existing data sources, even if one is a base map (Google Maps, in this case) and one is a "content" layer (such as the data scraped from Craigslist or the Chicago Police Department). Mashup creators pull data dynamically from one source and integrate it with another. The trend toward Mashups is already visible in other domains. In web terminology, a M*ashup* is a web site that combines ("mashes up") content from more than one source (from multiple web sites) into an integrated experience. Mashups are content aggregates that leverage the power of the Web to support worldwide sharing of content that conventionally would not have been easily accessible or reusable in different contexts or from different locations. I will call this the "**narrow**" definition of Mashups.

Mashups and tools to support the construction of Mashups are important to the development end-user programming [7]. Use of the web is pervasive for many tasks that involve searching for information or decision-making. Constructing Mashups typically involves programming, although there are now a number of tools that simplify or eliminate programming for a number of Mashup tasks.

Mashup development tools are often designed to address different Mashup patterns. With tools following what a different model Mashup is, what is meant by the term "Mashup" become unclear. Mashups originally referred to websites combining features of multiple websites. Some Mashup tools enable users to construct entire website as final artifact [6]. However, that definition has been blurred by tools that focus on data integration issues (such as Karma [9]). Also adoption of existing websites to alternative environments like mobile devices is the goal of a number of Mashups listed in the Mashups directory website. ProgrammableWeb.com as well as a design goal of the d.mix [4] web service integration tool.

There is an entire category of Mashups known as "Map" Mashups. Most likely owing to the inspiration from HousingMaps, these Mashups comprised over half of the Mashups listed on Programmable Web from 2005 to 2009. Many of these were simply a static data set or results form a query visualized on a map. In April 2007, Goggle introduced Google MyMaps, a feature of their mapping tool that enables users to plot customized data on a map without having to learn the Google Maps API and create website of their own. Subsequently as of May 2009 maps only compromise 46% of all Mashups listed on ProgrammableWeb[5]. Perhaps this is evidence that many maps Mashups are not Mashups at all, but simple uses of maps as visualization widgets, like charts, tables, and selection menus.

Since Mashups appear to follow various kinds of patterns and each Mashup Tool aim to support specific pattern, HenceI have propose that it's valuable to survey the space of Mashups and catalogue with those patterns use. Understanding patterns in Mashups can help drive the development of Mashup tools to support specific patterns.

This paper is structured as follows, In Section 2 presents the work carried out on the Survey, and Section 3 represents the outcome of the feasibility study and Section 4 discuses the results of the ongoing survey of Mashups presents and about

[3] On the original website of HousingMaps, Paul Rademacher called his project a "Craigslist-GoogleMaps combo site", avoiding the emotional connotations of "hack" or "mashup."
[4] http://www.mappinghacks.com

[5] www.programmableweb.com

the future of Mashup patterns and development tools.

2. Survey Methodology

In search of Mashup patterns exist, I qualitavely surveyed 22 Mashups from two Mashup directories that showcase Mashups. For each Mashup, I looked at what API were used (according to what each Mashup author claims in the directory), what data sources were involved, and what websites each Mashup used.

I also assigned category names to describe these improvements. For example, oSkope[6] provides an innovative visual search interface to Amazon and eBay that doesn't require users to visit multiple pages. It's also enables users to build up adhoc lists of items of interest, which is more difficult on Amazon and eBay where items need to be saved to a "wish list" triggering a page reload. Surprisingly, although the Mashup makes use of multiple websites, it doesn't unify data or integrate searches across them. I assigned this Mashup to the "Alternate UI category". I examined high–level user inputs and outputs. For example the "last.fm normalizer", accepts a last.fm user name and shows statistics about that user's playlist.

I created binary labels to describe interesting features in each Mashup. These labels were improved upon the original website and then checked to see if any Mashups I had already examined would deserve the label. Table 1 explains some of these labels.

Table 1. Some binary categories for each Mashup

Search	Is the Mashup a search interface?
Real-time	Is the purpose if the Mashup to allow the user to monitor or observe the original website as a real time data set?
Visualization	Does Mashup add visualization to the data?
Widget	Is the Mashup actually a widget for some platform like Apple's Dashboard, iPhone, or Google homepage?
Personalized	Does the Mashup make use of user's personal information from the original website, or enable the construction of a personalized data set form the original website?
Folksonomy	Does the Mashup make use of a tagging system or adds tagging to the original data set?
In-situ-use	Is the Mashup simply a tailoring of an original website optimized to a specific situation use?

It would be difficult to add this qualitative analysis to all Mashups in existence. On Programmable Web, there are over 3000 Mashups in its Mashup directory. Part of this is due to the fact that Mashup authors submit their own Mashups to the Programmable Web directory. However, many of these Mashups are poorly thought out, or assembled only for demonstration purposes and submitted to the directory to collect community feedbacks or to drive advertising revenue by being linked from a highly reputable site. In order to make this study feasible, I have examined a random sample of Mashups. To reduce the number of bad Mashup sample, I took a random of 22 Mashups that were in the set "Mashup of the Day" on Programmable Web. I also examined Mashups from Mashupawards.com, which accepts nomination of interesting websites and selects one Mashup each day to be given a "Mashup of the Day" award. Table 2 shows the selected Mashups for the survey.

Table 2: Selected Mashups

Mashup	APIs	Notes
Delimport Ianhenerson.org/delimport.org	delicious	Enable spotlight on Mac Os to index your delicious bookmarks.
EveryBlock www.everyblock.com	Yelp. Craiglist, Flickr infor feeds	Find out what's happening in your neighborhood, including official city buzz info
FlickrRealTime	Flickr	
HeyWhatsThat www.heywhatsthat.com	GoogleMaps, GoogleEarth, USGS, Elevation data	Complex featured Vista data, Print out what your view will be like at a particular summit
Leaflets www.getleaflets.com	Various	User alternative interfaces to various websites that are for iPhone
oSkope www.oSk	Amazon eBay	Not unifying data, provides details and links to

[6] www.oskope.com

ope.com		original pages, support ad-hoc explorations, still needs to customize by bringing necessary details up.
PackMapr Pack-mapr.com	UPS,USPS, FedEx, Google Maps	Enter tracking no of any service, see travel path, create an RSS feed that will change when the travel path changes
Song List www.song-list.net	iTunes, Napster, Rhapsody	Listing of artists names, aggregates the listing from 3 different sites
Soup-soup www.soup-soup.net	News sites, Flickr, wikipedia	Every news is appended with related photos and Wikipedia articles. Good for getting to know the background of the news topic
Tag-Browser www.tagbrowser.com	Flickr	Sort of faceted tagging style for sets of tags. Navigate along tags groups
Trendite www.trendite.com	Various Search engines	Monitor what's hot on searching
Vidmeter www.vidmeter.com	YouTube, MySpace, Google video	Show popular videos on major sites

3. Survey Outcome

The ad-hoc categories of Mashups as well as the labels can be taken as Mashup patterns. There are several categories and labels I found quite interesting. They categories are as follows:

Aggregation: A common function of Mashups is to aggregate multiple websites together or summarize set of data. But this takes no multiple forms. EveryBlock[7] takes and address and find nearby geo tags pictures from Flickr, finds local restaurants, finds reports of nearby crimes and other public notices, and looks for nearby Craig-List ads. This is similar to entering a single location as query into multiple websites and assembling the results onto a single page. Soup –Soup[8] is similar but uses keywords from new stories instead of single query terms to search Wikipedia and Flickr for relevant information. Vidmeter[9] finds the most popular videos from multiple websites and assembles them into a single raking chart. This probably requires some sorting and addition. Trendite[10] is a site that supports monitoring of search trends. It most likely make request for top search queries and results from the major search engines and assembles them onto a single page. Song List[11] assembles links to song samples and needs to unify song titles between multiple music services.

Alternate UI: These Mashups don't combine multiple websites at all but rather aim to support new methods of interacting with data from the website or support specific use cases. Leaflets[12] are specialized versions of common websites, such as upcoming.com, that are optimized to run on the Apple iPhone, thus saving users from having to visit the normal versions of those websites. TagBrowser provides an alternate oSkope supports searching and browsing better than Amazon and eBay's normal search tools by reducing page tools and support ad-hoc collection.

In-situ use: Mashups support the specialized use of a website outside of the typical use case. HeyWhatsThat[13] complies terrain elevation data to produce printable drawings that can help a person to determine what landmarks are visible from a vista point. Delimport is a plug – in for the del.icio.us bookmarking service that enable Mac users to access their bookmarks through a keystroke based search facility built into the operating system, instead of visiting the del.icio.us website.

Personalization: A number of Mashups personalize based on either personal information from the websites they are based on or new personal information from users. YES for eBay computes a year –end summary of sales and tax liabilities for a person's eBay account. Wishpot uses browser scripts to create personal waitlists encounter them on other websites. ExitAhead takes an artist list from a user's last.fm playlist and compile links to related paraphernalia.

Focused View of Data: In this category Mashup exists to index or categorize a subset of another websites entire content. YouTutorials is a user-submitted catalog of tutorials that can be found

[7] www.everyblock.com

[8] www.soup-soup.net

[9] www.vidmeter.com

[10] www.trendite.com

[11] www.song-list.net

[12] www.getleaflets.com

[13] www.HeyWhatsThat.com

on YouTube. Tags and a category system are further applied onto the set of video to facilitate searching.

Real –time Monitoring: A number of Mashups support real-time monitoring. TwitterWhere and FlickrRealTime show updates every few seconds, while Vidmeter might change only a few times an hour. EveryBlock's real time data comes from pictures, added restaurants reviews, and public notices, which are likely to see change on the scale of days.

I have presented a few interesting patterns that I have observed in this preliminary study. It is possible that there are more patterns in the Mashups I have encountered so far. **Table 3** shows Mashup patterns and what kinds of development tools required to build these Mashups. As we can see the development tools required vary with the Mashup patterns.

4. Discussion & Future Work

However the diversity of the forms of the Mashups in our data set requires us to examine a larger set in order to see the patterns for ourselves. Single data points don't make patterns.

I have noticed that single source Mashups tended to fit the Alternate UI or Real-time Monitoring pattern. Although most Mashup tools do not support the construction of complex user interface without dealing with HTML, there has been past work in building user situation. Apple's Safari web browser currently supports creating Dashboard widgets by selecting a DOM element from a web page that contains the data that will be of future interest. I expect to see more Mashups which are specialization interfaces.

Patterns come from high –level guesses of how Mashups make use of other websites and web services APIs. Mashup patterns may become richer when we look closely at the details of how data from other web sites and services are sued. Ideally, the ground truth in Mashup patterns can be found in the code itself, but it is unlikely that a survey looking at the source code of many Mashups will be feasible. In my final report I will be examining the APIs data that Mashups use more closely, reverse-engineering the Mashups in a sense.

In this study, I only looked at Mashups that are whole websites. However I have yet to look at Mashups or other artifacts that result from Mashup tools. These instances may not be complete Mashups in and of themselves but they will represent end –users' efforts towards customizing their web experiences.

We can conclude from our observations that Microsoft Popfly [6] and Marmite [11] conceive of Mashups as data flows that take data originating from a source or user input, applying web services to augment or transform the data, and visualizing the results on a map or on a web page. Yahoo Pipes [12] takes data from RSS feeds or web services and applies programming operations such as a text filtering, counting, sorting and a regular expression. Karma [9], a data-integration by demonstration system focus on data unification where disparate data sources are linked together by common attributes, similar to "joining" database tables. Enterprise Mashup tools, such as IBM's QEDWiki [5] focus on connecting graphical visualization to data sources, following a pattern established by the first Mashup, HousingMaps[14]. Also we can see that Intel MashMaker[15] can be used to develop most of the Mashups patterns that I observed in my survey. I can conclude that it's mostly because of the Intel MashMaker is the only client oriented system. The philosophy this tool follows *"mashing is just browsing"* and "mashup is personal" which contradicts the other development tools exist.

If we are to seek design targets for future Mashups tools, we should be looking at the artifacts resulting from existing tools as they will only show us the boundaries of what we can currently do. As we look into existing developed Mashups already present, programmers take past existing tools limitations, but we must also be aware the programming may not necessarily have the same needs as end-users. The more elusive targets are those applications that end-users have never considered and beyond, to applications that end-users didn't even think were possible.

References

[1] Erle, Schuyler. 2006. Web Map API Roundup. *Mapping Hacks*. April 7.

[14] www.housingmaps.com

[15] http://mashmaker.intel.com/

[2] Gamma, E, R Helm, R, Johnson, J, Vlissades. 1995. Design patterns elements of reusable object-oriented software. Addison-Wesley professional computing series. Reading, Mass: Addison – Wesley.

[3] Gibson, R., Erle, S., *Google Maps Hacks*. O'Reilly Media, 2006.

[4] Hartmann, B, L, Wu, K, Collins, and S, R, Klemmer, 2007. Programming by a sample: rapidly creating a web applications with d.mix. In Proc. Of UIST '07, 241-250. ACM

[5] IBM QEDWiki http://services.alphaworks.ibm.com/qedwiki

[6]Microsoft Popfly http://www.popfly.ms

[7] Nardi, B.A. 1983. A Small Matter of Programming. Cambridge; MIT Press.

[8] Singel, Ryan. 2005. Map Hacks on Crack. *Wired*, July 2.

[9] Tuchina, R. P. Szekely, and C.A. Knoblock, 2008. Building Mashups by Example, In Proceedings of IUI 2008: ACM Press.

[10] Van Duyne, D, K, J, A, Landay, and J.I. Hong, 2003. The Design of Sites Patterns, Principles, and Processes for Crafting s customer-Centered Web Experiences. Boston: Addison-Wesley

[11] Wong. Y. and I. Hong. 2007. Marmite: Towards End-User Programming for the Web. In Proc. Of CHI '07: ACM Press

[12] Yahoo Pipes.http://pipes.yahoo.com

Table 3: Mashup Patterns

Mashup	Category	Labels	Tools
Delimport Ianhenerson.org/ delimport.org	Alternate UI	Personalized, in-situ use, search	Intel Mashmker,
EveryBlock www.everyblock.com	News, Aggregation, personalized, filtering	Real – time, Personalized	Yahoo Pipes, Intel Mashmaker, QEDWiki, Ms Popfly
FlickrRealTime	In Real-time		
HeyWhatsThat www.heywhatsthat.com	visualization	Search, Visualization In-situ	Ms Popfly, Marmite,
Leaflets www.getleaflets.com	Alternate UI	Widget	Intel Mashmaker, Ms Popfly
oSkope www.oSkope.com	Richer interaction, Alternate UI	Search, Visualization	Yahoo Pipes, Karma, Intel Mashmaker
PackMapr Packmapr.com	Simplification, Notification	Visualization, Real - time	Yahoo Pipes, Intel Mashmaker, QEDWiki
Song List www.song-list.net	Aggregation	Search	Intel Mashmker,
Soup-soup www.soup-soup.net	News, Augmentation		Intel Mashmaker
TagBrowser www.tagbrowser.com	Alternate UI	Search, Folksonomy	Ms Popfly, Marmite, Intel Mashmaker
Trendite www.trendite.com	Monitoring, aggregation	Search	Ms Popfly, Marmite, Intel Mashmaker
Vidmeter www.vidmeter.com	Aggregation, Summarization	Search, Real-time	Ms Popfly, Marmite, Intel Mashmaker

Table 3: Mashup Patterns