

## Toward a New Paradigm: Mashup Patterns in Web 2.0

<sup>1</sup> Cheng-Jung Lee, Ph. D. Student of Department of Information Management,  
National Yunlin University of Science and Technology, Taiwan R.O.C.  
Email: [rabbit66@ms61.url.com.tw](mailto:rabbit66@ms61.url.com.tw)

<sup>2</sup> \*Shung-Ming Tang, Professor, Department of Information Management,  
National Yunlin University of Science and Technology, Taiwan R.O.C.  
Email: [tangsm@yuntech.edu.tw](mailto:tangsm@yuntech.edu.tw)

<sup>2</sup> Chang-Chun Tsai, Associate Professor, Institute of Management, MingDao University  
MingDao University, No. 369, Wen-Hua Rd., Peetow, ChangHua 52345, Taiwan, R.O.C.  
Email: [jimtsai@mdu.edu.tw](mailto:jimtsai@mdu.edu.tw)

<sup>3</sup> Ching-Chiang Chen, Ph. D. Student of Department of Information Management,  
National Yunlin University of Science and Technology, Taiwan R.O.C.  
Email: [mw454278@gmail.com](mailto:mw454278@gmail.com)

*Abstract:* The advent of Web 2.0 changes the playing field drastically and opens a new era of platform opportunities. By opening the platform and making it remix, Google Maps changed the rules. Now, the rest of the industry is struggling to catch the world of web Mashups and Google's momentum. However, what about the conceptual heart of other popular Mashups? We focus on looking for patterns in successful Web 2.0 Mashups. We systemizes concepts from Top 60 popular Mashups on ProgrammableWeb— and aim to present seven collection of Mashup Patterns: Data source Mashups, Process Mashups, Consumer Mashups, Enterprise Mashups, Client-Side Mashup, Server-Side Mashups and Developer assembly Mashups which build platforms to foster innovation in assembly, where remixing of data and services create new opportunities and markets. In this paper we highlight latest Mashup trends, exploration of the design principles of Mashups architecture, evaluation the most popular Mashups technology for the enterprise level, seven options patterns that exist for implementation, and capture the fundamental behavioral aspects of Mashups. What we present in this paper can be generalized for other Mashups.

*Key-Words:* - Mashup, Web 2.0, Pattern, paradigm, Google Maps, API

### 1 Introduction

With Web 2.0 technologies spreading rapidly across the Internet, Mashups have recently attracted much attention as a promising approach for ad-hoc information and service integration. [22] When the Google Maps API was released in June 2005, there was an explosion of location-related software development. Craigslist apartment listings, Chicago crime data, Flickr photos, current news events, happy-hour locations, weather, historical sites, public transportation...just about any piece of content imaginable could be overlaid on a map using this API with a little JavaScript. [14]

Mashups have become an extremely popular way for developers to access and play in the Rich Internet Application world. As developers combine existing services with user experiences of their own

design, entirely new applications emerge. Another factor that has helped make Mashups popular is, Web browsers offer better support for Ajax, which implies increased speed. Also, open source software has grown more popular. The implication is that many more people are getting involved in developing contents that can be used by the general public. By opening the platform and making it remix, Google Maps changed the rules. Now, the rest of the enterprise is struggling to catch Google's momentum and the world of web Mashups. However, what about the conceptual heart of other popular Mashups? Despite rapidly increasing interest in Mashups over the past two years, comprehensive frameworks are lacking.

Since Mashups appear to follow various types of patterns and each Mashup tool aims at supporting specific patterns, we propose that it is worthwhile to

survey the space of Mashups and classify what those patterns are. The idea of patterns has been used in the past to present generalized solutions to problems that may be encountered when designing websites or writing software [7, 23]. Understanding patterns in Mashups can help drive the development of Mashup tools to support specific patterns. [11] To our knowledge, there has been very little survey of Mashup patterns. In this paper, we describe a preliminary survey we conducted and report on observed patterns.

There are some existing classifications of Mashups in various literatures available on this subject. [17] Many supplier and industry analysts talk about Mashups in various ways. In many cases, web Mashups are categorized according to their functionality; for example, some define data Mashups, business Mashups, photo and video Mashups and news Mashups. However, we classify web Mashups by seven major categories which are often used in the enterprise. We focus on looking for patterns in successful Web 2.0 Mashups. We systemize concepts from Top 60 popular Mashups—and aim to present seven collection of Mashup Patterns: (1) Data source Mashups (2) Process Mashups (3) Consumer Mashups (4) Enterprise Mashups (5) Client-Side Mashups (6) Server-Side Mashups and (7) Developer assembly Mashups which build platforms to foster innovation in assembly, where remixing of services and data creates new markets and opportunities. Although a slightly differences of these types of capabilities exist, the patterns we present in this paper and this discussion provide a quick method for classifying Mashup efforts.

As of MashupFeed (<http://Mashupfeed.com>), a Mashup advertisement and aggregation site, is adding more than two new Mashups a day from contributors—some incredibly cool, some inane, some destined to become winners in the Web 2.0 space, and some losers. Therefore, specifically, we overview some of the most popular Mashup tools and show how they facilitate the development of Rich Internet Applications. In this paper we highlight latest Mashup trends, exploration of the design principles of Mashups architecture, evaluation of the most popular Mashups technology for the enterprise level, seven options patterns that exist for implementation, and capture the fundamental behavioral aspects of Mashups. What we present in this paper can be generalized for other Mashups.

## 2 Related work

### 2.1 What is Web 2.0

Web 2.0, refers to a perceived or proposed second generation of Internet-based services—such as social networking sites, wikis, communication tools, Mashups and folksonomies—that emphasize on online collaboration and sharing among users. [15] Web 2.0 has also been called the social web, because its content can be more easily generated by users, as well as the collective intelligence of users. Users are not the passive consumers of content, but co-producers. Interaction plays an important role in Web 2.0 to create shared information.

Web 2.0 is not a uniform concept, but a generic term or metaphor for new Internet technologies and applications. Web 2.0 can be seen as a revival, intensification, renewal in which user generated content has a central place. [13] Osimo and Burgelman [16] state that Web 2.0 is about both technology and attitude. There is some debate how new Web 2.0 really is. Tim Berners-Lee for one, the founder of the WWW, views Web 2.0 merely as the logical further development. In that sense there is no new software or application with the name of Web 2.0. And yet what we observe that Web 2.0 Mashup is a paradigm shift how users use the web, a development that questions everything that has been developed and applied so far.

Below is the list of Web 1.0 versus Web 2.0 examples. (See Fig. 1) [10] The figure compares what the old Web had to offer to the more recent evolution.

Web 1.0	→	Web 2.0
DoubleClick	→	Google AdSense
Ofoto	→	Flickr
Akamai	→	BitTorrent
mp3.com	→	Napster
Britannica Online	→	Wikipedia
personal websites	→	blogging
evite	→	upcoming.org and EVDB
domain name speculation	→	search engine optimization
page views	→	cost per click
screen scraping	→	web services
publishing	→	participation
content management systems	→	wikis
directories (taxonomy)	→	tagging ("folksonomy")
stickiness	→	syndication

Fig. 1 The list of Web 1.0 versus Web 2.0 examples

In the world of Web 2.0, web sites are no longer stand-alone entities. Instead of simply displaying their wares to passing visitors, they become data centers—feeding information to other applications on the web. The information is not only shared, it is enriched. Users of shared data are encouraged to add to it, to annotate it. They identify points of interest on Google Maps, add tags to

photos on Flickr, and write book reviews on Amazon.com. Users help identify connections between pieces of data—they place their photos on maps of the world, they tag related links in del.icio.us and they create lists of related items on Amazon.com. [2]

## 2.2 Web Mashups

A web Mashup is a new type of web application that uses data and services from one or more external sources (usually from the Internet) to build entirely new and different web applications. [3] A Mashup is a more user-facing and user-directed process, where an end user directs an application to combine and aggregate data from various sources and combines the data with a presentation to provide a comprehensive view of the aggregation. Such a Mashup provides great value to the end user. [1, 8, 24] Data feeds such as RSS and ATOM feeds have been around for a while, making information available for anyone to re-use in another application. What's so different about web Mashups? The answer is that while older data and service aggregation technologies aggregate and integrate in a fashion, a true web Mashup creates a completely different and new function out of the existing content and services, driving different purposes and objectives. [2]

A Mashup is a Web page that uses Web 2.0 technologies, which may include JavaScript, PHP [19], and XML, to present information from a variety of sources or in a variety of ways where the presentation enhances the information. [21] The idea behind web Mashups is creativity and innovation in new data and services, not just aggregation of existing ones, which most of the older technologies focus on. [2] Mashups, more specifically called web application hybrids by Wikipedia, have been an exciting trend in web applications in recent years. [3] Web Mashups represent a new way of developing software and along with any new development techniques and coming opportunities.

## 2.3 Application Programming Interface (API)

Web APIs are application programming interfaces that are available over the Internet. They are also sometimes referred to as Web services. Web APIs are typically a related collection of Web services. It is helpful to think of a Web API as a series of Web services, each of which has one or more procedures that can be called using the Internet. An example of a Web API includes the Google API, which makes various search functions available over the Internet for use in third-party applications. Although Web

APIs and Web services are separate concepts, it is common and acceptable to refer to the two concepts interchangeably because they are so closely related. [5] Most Web APIs have a free limited-use license or trial period, and some of them require payment for the service.

## 3. SURVEY METHODOLOGY

To come up with some Mashup patterns, we qualitatively surveyed the most popular Mashups from ProgrammableWeb. [20] (ProgrammableWeb lists over 4200 Mashups as of 2009-07-15). For each Mashup, we studied what APIs were used, what context were involved, what tag is it and what websites each Mashup used. We also attempted to make some guesses at what purpose the Mashup might serve or how each Mashup improved on existing websites either user experience or in terms of functionality. In order to reduce the number of bad Mashups and make this study feasible, we took samples of the Top 60 Mashups by popularity ranking on Programmable Web. (Due to the limit of space, we only list 20 of them, please see selected Mashups and patterns in Appendix A)

### 3.1 The API Scorecard

Yahoo, Google, Amazon, eBay, and Microsoft have published application programming interfaces (APIs) based on web standards that allow you to utilize their complicated functionality without being a programming expert. Dozens more companies, big and small, have followed in the same way, creating a Mashup explosion of API mixing and matching. New, sometimes strange, Mashup creations pop up all the time. This is help to understand how the web itself is becoming the next development platform. On the X axis are 5 major API vendors, they are sometimes referred to as GAMEY. There is also an Other column. On the Y axis are the primary types or categories of APIs being offered. If at any Vendor/Type intersection there is an API, it will display a checkmark and show the number of Mashups using that API as cataloged at ProgrammableWeb (A subset of all that exist). Many of the industry's largest players are creating new, competing APIs. This tool allows you to visualize the major APIs, compare them, and drill-down to learn more about them.[17] (See Fig.2)

Categories of APIs	Google	Amazon	Microsoft	eBay	Yahoo!
Advertising	• 24		• 0		• 4
Answers					• 19
Blogs	• 22		• 4		
Bookmarks					• 137
Chat	• 8		• 19	• 27	• 3
Classifieds	• 45		• 5		
Database		• 6			
Desktop	• 12		• 3		• 13
Events	• 31				• 33
Hosting		• 44			
Identity	• 4		• 0		• 3
<b>Mapping</b>	<b>• 1768</b>		<b>• 173</b>		<b>• 129</b>
Music					• 2
Office	• 12	1768 mashups listed here			
Payments	• 7	• 2		• 15	
Photo	• 19		• 1		• 463
Presence					
Search	• 89	• 6	• 31		• 124
Shopping		• 309		• 175	
Social	• 3				
Storage		• 51			
Video	• 399				• 21
Voice	• 8			• 27	• 3
Widgets	• 89		• 4		• 13
All	54/2463	17/405	26/232	6/240	35/892

Fig. 2 The API Scorecard

### 3.2 Most popular API ranking by Mashups

An API is a set of protocols, routines, and tools for building software applications which is like a no-frills website, accessed through URL requests but returning structured data instead of web pages displayed in a browser. A good API makes it easier to develop a program by providing all the building blocks. By combining several kinds of requests, you can use an API to power your own custom applications. A programmer then puts the blocks together. The following are the Top 10 APIs for Mashups: (See Fig. 3)

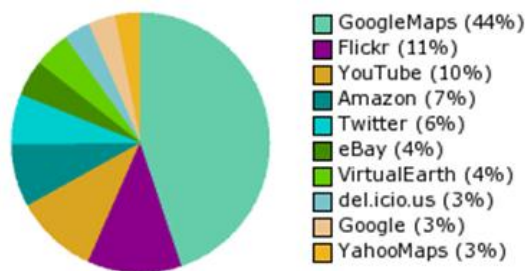


Fig.3 Top10 APIs for Mashups [18]

For example, the most popular Mashup APIs as follows:

#### Top1: Google Maps

An API defines a standard way for one program to call code that lives within another application or library. The Google Maps API using a simple JavaScript interface or a Flash and makes it easy to put a map on your own web page. The API includes language localization and geocoding, and has mechanisms for enterprise developers who want to utilize the Google Maps API within an intranet.

#### Highlights (Google Maps):

Summary: Mapping services

Category: Mapping

Tags: mapping

Protocols: JavaScript

Data Formats: XML, VML, JSON

API home:

<http://code.google.com/apis/maps/index.html>

This is 1 of 54 APIs by Google

#### Functionality (Google Maps):

- API Groups: Uses a JavaScript object model
- Example API Methods: GMap2, GMarker, GPolyline, GIcon, GEvent, GBounds, GSize, GClientGeocoder

#### Top2: Flickr

The Flickr API can be used to retrieve photos from the Flickr photo sharing service using a variety of feeds - public photos, videos, favorites, friends, group pools, discussions, and more. You can use the Flickr API to build a gallery web site displaying a selection of your recent photos. The API can also be used to upload photos and video. The Flickr API supports many protocols including REST, SOAP and XML-RPC. Most applications built around Flickr use one or another of the popular API kits to take care of the rather tedious business of constructing request messages and parsing responses.

#### Highlights (Flickr):

Summary: Photo sharing service

Category: Photos

Tags: photo video

Protocols: REST, SOAP, XML-RPC

Data Formats: XML, JSON, PHP

API home: <http://code.flickr.com>

This is 1 of 39 APIs by Yahoo.

**Functionality (Flickr):**

API Groups: auth, blogs, contacts, favorites, groups, interestingness, people, photos, photosets, reflection, tags, test, urls

Example API Methods: photos.addTags, people.getInfo, tags.getRelated

Service Endpoint: <http://api.flickr.com/services/rest/>  
And <http://api.flickr.com/services/soap/>

**Top3: YouTube**

Their new GData API allows user to do all the things they could with the old API. They have added more flexibility with filtering, querying, and alternative outputs in JSON. Brand new feeds include video responses, full comments list, and related videos. Additionally, the Google data client libraries are at your disposal.

**Highlights (YouTube):**

Summary: Video sharing and search

Category: Video

Tags: media video

Protocols: GData, Atom Publishing Protocol

Data Formats: Atom, RSS

API home: <http://code.google.com/intl/zh-TW/apis/youtube/overview.html>

This is 1 of 54 APIs by Google

**Functionality (YouTube):**

- API Groups: Video feeds, video uploads, User playlists, profiles, subscriptions Video comments
- Example API Methods: VideoFeeds
- Service Endpoint: <http://gdata.youtube.com/feeds/>

We found that all APIs described the functionality provided, the messaging formats supported, the protocols and the programming languages they support (known as programming language bindings), for examples, the Most Popular Mashup APIs— **Google Maps**. The Google Maps API allow for the embedding of Google Maps onto web pages of outside developers, using a simple JavaScript interface or a Flash interface.

Tags find Mashups and APIs by user-defined keywords. The following is a tally of tags for Mashups recorded on programmableweb indicates recent Mashup trends. (See Fig. 4)

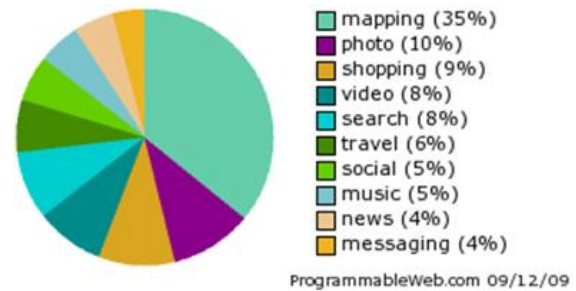


Fig. 4 Top Mashup Tags

Table 1 APIs are ranked based on their popularity

API	Description	Category	Mashups
Google Maps	Mapping services	Mapping	1768
Flickr	Photo sharing service	Photos	463
YouTube	Video sharing and search	Video	399
Amazon eCommerce	Online retailer	Shopping	308
Twitter	Microblogging service	Social	235
eBay	Online auction marketplace	Shopping	175
Microsoft Virtual Earth	Mapping services	Mapping	173
del.icio.us	Social bookmarking	Bookmarks	137
Google Search	Search services	Search	134
Yahoo Maps	Mapping services	Mapping	129
Yahoo Search	Search services	Search	124
411Sync	SMS, WAP, and email messaging	Messaging	120
Last.fm	Online radio service	Music	116
Facebook	Social networking service	Social	102
Google Ajax Search	Web search components	Search	89
Google Homepage	Portal gadgets	Widgets	89
Yahoo Geocoding	Geocoding services	Mapping	88
Technorati	Blog search services	Blog Search	67
GeoNames	Geographic name and postal code lookup	Mapping	57
Amazon S3	Online storage services	Storage	51
Yahoo Image Search	Image search services	Search	50

### 3.3 Web protocols

A tally of protocols Usage by APIs indicates recent Protocol trends. (See Fig. 5) Both SOAP and REST are platform neutral protocols for communicating with remote services. As part of the service-oriented architecture paradigm, clients can use SOAP or REST to interact with remote services without knowledge of their underlying platform implementation: the functionality of a service is completely conveyed by the description of the messages that it requests and responds with.

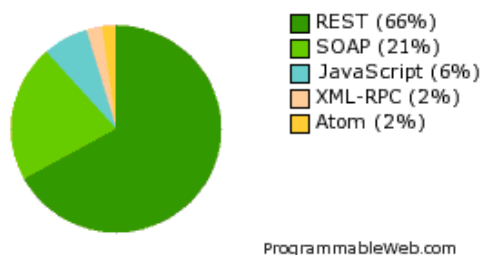


Fig.5 Protocol Usage by APIs

### 4. SURVEY FINDINGS

Our patterns come from how Mashups make use of other websites and various APIs. A Mashup can be viewed as being composed of three different participants, which are usually physically separated too. They are (1) Content providers (2) Mashup site and (3) Client's browser. Please refer to our findings in Appendix A, where APIs categories of Mashups as well as the tags can be taken as Mashup patterns. Our survey provides seven major Mashup patterns which are often used in the enterprise:

#### Pattern 1: Data source Mashups

Data source Mashups combine similar types of media and information from multiple sources into a single representation, such as combining the data from multiple RSS feeds into a single feed with a graphical front end. Normally it can integrate and combine information from the following sources: remote Web services/XML, REST, RSS/ATOM, Content Management Systems (CMS), databases, spreadsheets, text and PDF files or even just plain HTML (web pages) and combines it with data from another source.

An example of a data Mashup is the travel site <http://www.kayak.com>. Kayak is a comprehensive travel search engine which gets its data from over 100 other travel sites. Kayak therefore does not sell directly to customers but serves as a portal through which customers can be directed to travel agencies that can serve their needs. An enterprise data source

Mashup usually integrates data from internal and external sources. For example, it could create a market share report by combining an external list of all houses sold in the last week with internal data about which houses one agency sold.

Please see Mashup architecture (web data aggregation) as follows: (Fig. 6)

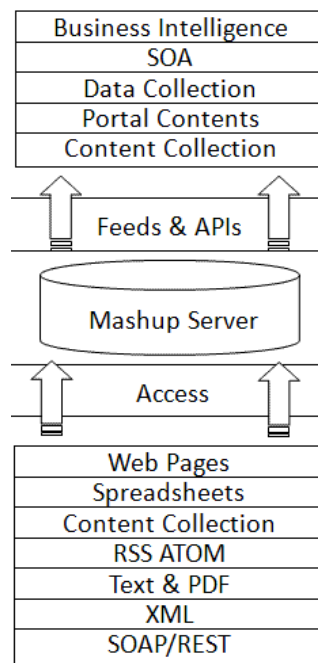


Fig. 6 Mashup architecture

There are many tools on the market that allow data Mashups to be created in a simple manner. For example, Yahoo Pipes [23] is a web application that permits users to build aggregate web feeds by integrating data from different sources using a very intuitive graphical user interface. Microsoft offers a very similar application, called Microsoft PopFly [6]. The Google Mashup Editor [9] is another on-line application used for creating such Mashups.

The following are features to make a data source good for relevant web mapping applications:

- Useable data: The data is free and open and the data's license does not limit its use.
- Community data: All users can contribute new data and modify existing data.
- Organized data: The data's explicit structure conveys its meaning and its structure unifies the whole collection.
- Accessible data: The data source has a web-based query interface which publishes and tracks all changes.

- A foreseeable future: The service is built on scalable technology and the company running the service with a foreseeable future.

**Pattern 2: Process Mashups**

Process Mashups deemed more appropriate for repeatable processes. At its core, it is about a level of agility into the process of stringing together services into an application, or Service-oriented business applications (SOBAs). Often called SOA orchestration, Mashup assemblers (developers or technical business users) leverage a set of existing Business Process Execution Language (BPEL) processes and connect them together in the context of their required task. In this architecture (see Fig. 7), different user groups can combine UI functions as they see fit, without having to create many variations of the same UI. This is critical for the successful deployment of a process Mashup.

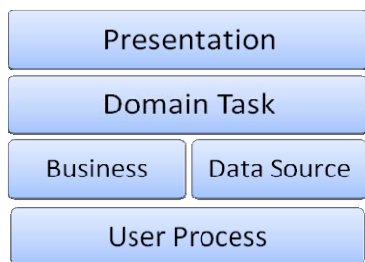


Fig. 7 Process Mashup Architecture

**Pattern 3: Consumer Mashups**

The most common type of Mashup is the consumer Mashup, aimed at the general public. A consumer Mashup is an application that combines data from multiple public sources in the browser and organizes it through a simple browser user interface. A good example of a consumer Mashup is HousingMaps (<http://www.housingmaps.com>), which gets rental listings from Craigslist and displays these listings on a Google Map for a visual representation of local apartments for rent by using Google Maps' API. The results displayed below are searches for any bedroom apartments in Houston that cost between \$750 and \$1250 which is powered by craigslist and Google Maps. (See Fig. 8 and Fig. 9)

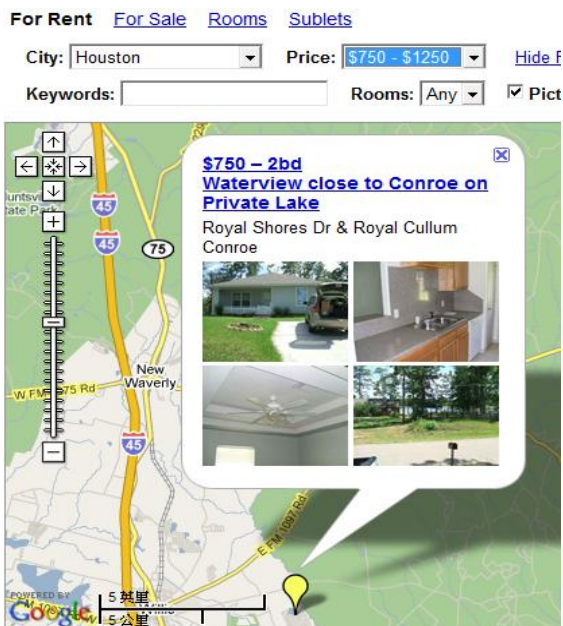


Fig. 8 An example of a consumer Mashup

pics	price	bd	description	city	date
	\$889	1bd	<a href="#">Upscale Community with First Class Amenities</a>	Houston	9/14
	\$982	1bd	<a href="#">Beautiful apt homes, must see!</a>	Houston	9/14
	\$1200	1bd	<a href="#">Live in your dream home at Camden Royal Oaks</a>	Houston	9/14
	\$917	2bd	<a href="#">Your apartment search stops here!</a>	Stafford	9/14
	\$1036	2bd	<a href="#">Fabulous 2 Bedroom Apartment Home</a>	Webster	9/14
	\$1080	3bd	<a href="#">Rent to Own! Nice Neighbors! Ez Terms</a>	Houston	9/14
	\$859	2bd	<a href="#">Beautiful 2 Bedroom ready for you to move in today</a>	Spring	9/14
	\$913	2bd	<a href="#">One Bedroom with Study</a>	League City	9/14
	\$975	1bd	<a href="#">Gorgeous views are waiting at Camden City Centre</a>	Houston	9/14
	\$829	1bd	<a href="#">Amenities you Deserve @ prices you love! Spacious apartments+much more</a>	Houston	9/14
	\$859	1bd	<a href="#">Great Location</a>	Houston	9/14

Fig. 9 An example of a consumer Mashup

**Pattern 4: Enterprise Mashups**

An enterprise Mashup, also referred to as a business Mashup, is an application that combines data from multiple internal source of data (a customer service alerts blog and a database of service locations) and public sources (such as weather and news) and publishes the results to enterprise portals, application development tools, or as a service in an SOA cloud. Similar to consumer Mashups, but solve business problems. Many of these early Mashups were consumer-focused, although recently there has started to be both interest and acceptance of Mashups in the enterprise. Organizations are

starting to realize that they can put their well-defined services that do discrete bits of business logic together with other existing services, internal or external to the organization, to provide new and interesting views on the data.

Although Mashups started as a consumer phenomenon, some credible examples bring the advantages of Mashup while addressing corporate concerns over security, governance, and enterprise-ready architecture. Mashups can help corporations embrace Web 2.0 ideals by leveraging existing SOA investments and improving time to market for online services. Web 2.0 technologies such as wikis, tags, and Mashups are increasingly being used to breathe new life into corporations, and this trend has been dubbed “Enterprise 2.0.” An enterprise Mashup can contain anything important to a corporation and can contribute directly or indirectly to its bottom line. Enterprise Mashups can affect all parts of the business from information workers to sales and management. Mashups help maintain data integrity and facilitate the use of one view of a system: [12]

Companies will also combine information from enterprise search engines, web services, messaging systems, business intelligence engines and data integration solutions and combine that information from external services from their partners and emerging external data sources to deliver the information up to the glass. Enterprise Mashup uses the assets of the Web and Enterprise for Ad Hoc self-service application creation. (See Fig. 10)

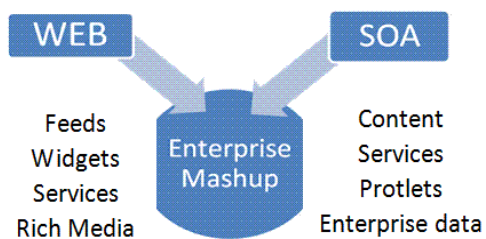


Fig. 10 Enterprise Mashup uses the assets of the Web and Enterprise

Enterprise Mashup adoption is driven by the need to transform the economics of integrating applications to make people more efficient. Enterprise Mashups must also interoperate with enterprise application technologies for security, governance, monitoring, and availability. Many enterprises are embracing Mashups for various reasons. Some need their software systems to change often to keep up with the rapid rate at which their business needs change. Business Mashups

must be able to access the back-end business systems—such as finance, HR, and CRM—on which organizations run. With business Mashups, domain experts use interfaces to back-end systems to build smaller point applications.

**Pattern 5: Developer assembly Mashups**

Mashups are built using different programming languages such as Perl, PHP, .Net, and others to call URL-based or web-based Representational State Transfer/Plain Old XML (REST/POX-based) services and link them together.

**Pattern 6: Client-Side Data Mashup**

Client-Side Mashups integrate services and content on the client. They mash up directly with the other web site's data or functionality. New information that didn't exist before can result such as when addresses are geocoded and display on a map to create a visualization that could exist without the underlying combination of data. For example, in a Client-Side Mashup, the client might make requests directly to the other web site. As for Client-Side Software Mashup, this is where code is integrated in the browser to result in a distinct new capability. While a component model for the browser is only now being hashed out as part of Open Ajax, there is considerable potential in being able to easily wire together pieces of browser-based software into brand new functionality. In general, a Client-Side Mashup works as illustrated in Figure 11.

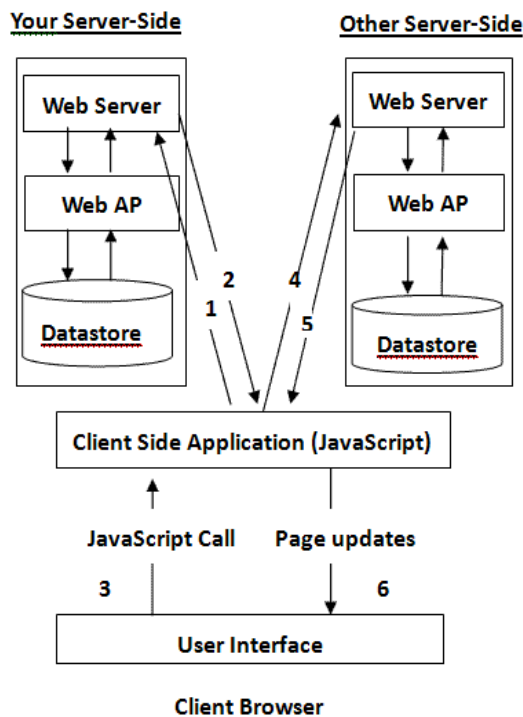


Fig. 11: How a Client-Side Mashup Works



1. HTTP Request: The browser makes a request to the server in your web site for the web page.
2. HTTP Response: The server on your web site loads the page into the client. The page usually includes a JavaScript library from the Mashup site.
3. Call function: Some action in the browser page calls a function in the JavaScript library provided by the Mashup site, or it calls the custom JavaScript function that you created.
4. HTTP Request: Several kinds of HTTP request methods are possible, but most web pages deal with just two: GET and POST. A request is made to the Mashup site to load the script.
5. HTTP Response: A callback allows a program to pass a reference to a dynamic function on the application side. The Mashup site loads the script. Typically, a local callback in the browser page is executed with a JavaScript Object Notation (JSON) object sent as a parameter. The callback is a development feature that allows better integration with client-side applications.
6. Page Updates: The callback function updates the client side of the page.

### Pattern 7: Server-Side Mashups

Mashups aren't just browser-based integration, but can be Server-Side too. Databases have been linking and connecting data for decades, and as such, they have relatively powerful mechanisms to join or Mashup data under the covers, on the Server-Side. Server-Side Mashups integrate services and content on the server. In a Server-Side Mashup, the work is pushed from the web application client to the server. A significant reason for using the Server-Side style is to overcome browser security restrictions that constrain XMLHttpRequests and the ability to access web sites across domains. A Server-Side Mashup needs to send several simultaneous HTTP requests to partner services for each inbound request. A Server-Side Mashup is running on a machine, and it consumes two other partner services to generate responses. As for Server-Side Software Mashup, recombinant software is probably easier right now on the server since Web services can more easily use other Web services and there are less security restrictions and cross domain issues. As a result, Server-Side Mashups like those that in turn use things like Amazon's Mechanical Turk or any of the hundreds of open Web APIs currently available, are quite common.

The Pet Store Mashup with the Yahoo Maps Geocoding service is an example of a Mashup that uses the Server-Side style. Recall that Pet Store uses the Yahoo Maps Geocoding service together with

the Google Maps service to display a map of pet locations. Pet Store combines its own data and functionality with latitude and longitude data returned from the Yahoo Geocoder service for seller addresses and with maps from the Google Maps service. The result of these three applications mashing up together is a page where users can use a map to browse for pets. In general, a Server-Side Mashup works as illustrated in Figure 12.

A Server-Side Mashup works as follows:

1. JavaScript Call: A user generates an event in the client, typically a web page in a browser. The event triggers some JavaScript function on the client side.
2. HTTP Request: The client makes a request to the server in your web site, typically an Ajax request in the form of an XMLHttpRequest object. The XMLHttpRequest object can be used by scripts to programmatically connect to their originating server via HTTP. The XMLHttpRequest specification defines an API that provides scripted client functionality for transferring data between a client and a server.
3. Web server receives the request and calls a method on a Java class or on multiple Java classes to perform the action.
4. HTTP Request: The proxy class processes the request and opens a connection to the server of the Mashup site, that is, the web site provides the needed service. A proxy class is basically a class that implements a list of interfaces specified at runtime such that a service method invocation through one of the interfaces on an instance of the effective class will be encoded and dispatched to another object through a uniform interface.
5. HTTP Response: The Mashup site receives the request, usually in the form of a GET or POST, processes the request, and returns data to the proxy class.
6. Data Transfer: The proxy class receives the response and may transform it in a format appropriate to its data type for the client.
7. HTTP Response: The servlet returns the response to the client.
8. Page Updates: A callback function exposed in the XMLHttpRequest updates the client side page by manipulating the Document Object Model (DOM) that represents the page. The DOM is the set of objects that represent the page. Manipulation of the DOM (which is possible through JavaScript) causes changes or updates in the document.

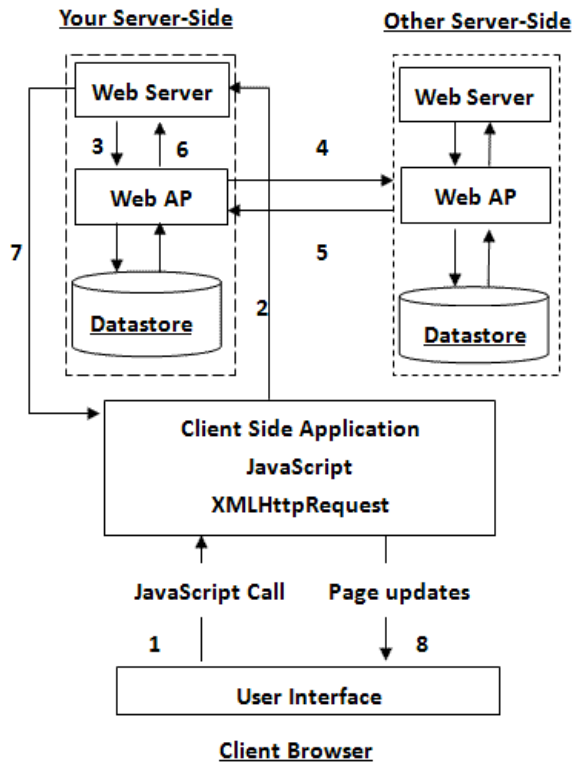


Fig. 12: How a Server-Side Mashup Works

## 5. Conclusions

We have presented seven interesting patterns that we have observed so far in this study. This paper describes valued preliminary work in the uncovering of Mashup patterns in order to find a new way of providing new web services and new directions for the design of Mashup tools. Our patterns come from how Mashups make use of data sources, other websites and various APIs. It is possible that there are more patterns exist in the Mashups than what we have encountered so far. Future work may examine the APIs and data that Mashups use more closely. These instances represent users' efforts towards customizing their website experiences.

With this overview of Top 60 popular of Mashups, we argue that not only is the web itself become a platform, individual web sites are becoming platforms and platform components as well. Platform solutions and ecosystems exist for the purposes of sharing information. Each connection made is stored away—an extra data point is created. By encouraging both the sharing and the enhancement of data, the overall value of those data is increased. We present the main Mashup patterns behind the concepts of web2.0. The most popular Mashups and the most popular APIs are summarized and recent protocol trends are

introduced. What we observe with Web 2.0 Mashup is a paradigm shift how users use the web, a development that questions everything that has been developed and applied so far. This calls for more research that can improve our knowledge on the Mashup strategies, business models and relevant technologies involved.

### References:

- [1] Chang-Chun Tsai, Cheng-Jung Lee, Shung-Ming Tang, The Web 2.0 Movement: Mashups Driven and Web Services, WSEAS TRANSACTIONS on COMPUTERS, Issue 8, Volume 8, August 2009, pp.1235-1244
- [2] Chang Sau Sheong, Ruby on Rails Web Mashup Projects, Packt Publishing, ISBN 978-1-847193-93-3, 2008
- [3] Chris Korhonen, David Hassoun, John Crosby, Creating Mashups with Adobe Flex and AIR, Friends of ED, ISBN-13: 978-1-59059-936-5
- [4] David Wilkinson, Flickr Mashups, Wiley Publishing, ISBN: 978-0-470-09774-8, 2007
- [5] Denise Gosnell, Professional Web APIs: Google®, eBay®, Amazon.com®, MapPoint®, FedEx®, Wiley Publishing, 2005, ISBN-10: 0-7645-8445-6
- [6] Microsoft PopFly, <http://www.popfly.com>
- [7] Gamma, E., R. Helm, R. Johnson, J. Vlissades. Design patterns elements of reusable object-oriented software. Addison-Wesley professional computing series. Reading, Mass: Addison-Wesley, 1995.
- [8] Fan Yang, Zhi-Mei Wang, A Mobile Location-based Information Recommendation System Based on GPS and WEB2.0 Services, WSEAS TRANSACTIONS on COMPUTERS, Issue 4, Volume 8, April 2009, pp. 725-734
- [9] Google Mashup Editor, <http://editor.googleMashups.com>
- [10] James Governor, Dion Hinchcliffe, and Duane Nickull, Web 2.0 Architectures, O'Reilly, 2009, ISBN: 978-0-596-51443-3
- [11] Jeffrey Wong, Jason Hong, WEUSE IV, May 12, 2008, Leipzig, Saxony, Germany, ACM 1-58113-000-0/00/0004
- [12] John Shaw and Simon Evans, Pro ADO.NET Data Services: Working with RESTful Data, Apress, 2009, ISBN-13 (pbk): 978-1-4302-1614-8
- [13] Johan van Wamelen and Dennis de Kool, Web 2.0: A Basis for the Second Society?, ICEGOV2008, December 1-4, 2008, Cairo, Egypt, ACM 978-1-60558-386-0/08/12

- [14] Michael Young, Google Maps Mashups with Google Mapplets, FirstPress, ISBN: 9781430209959, 2008
- [15] Miltiadis D. Lytras (Editor), Web 2.0 The Business Model, Miltiadis D. Lytras, Springer Science+Business Media, LLC. ISBN-13: 978-0-387-85894-4, 2009
- [16] Osimo, D., Burgelman, J.C., Web 2.0 for e-Government: Why and How, 4th Ministerial e-Government Conference, Lisbon, 2007
- [17] Programmableweb,  
<http://www.programmableweb.com>
- [18] Programmableweb API Dashboard,  
<http://www.programmableweb.com/apis> ,  
Accessed Sep 12, 2009
- [19] Shu-Wai Chow, PHP Web 2.0 Mashup Projects, PACKT publishing, ISBN 978-1-847190-88-8, 2007
- [20] Todd Stauffer, How to Do Everything with Your Web 2.0 Blog, McGraw-Hill Companies, ISBN: 9780071492188, 2008
- [21] Tuchinda, R., P. Szekely, and C. A. Knoblock. Building Mashups by Example. In Proceedings of IUI 2008: ACM Press, 2008.
- [22] Van Duyne, D.K., J.A.Landay, and J.I. Hong, The design of sites patterns, principles, and processes for crafting a customer-centered Web experience. Boston: Addison-Wesley, 2003.
- [23] Yahoo Pipes, <http://pipes.yahoo.com>
- [24] Zhi-mei Wang, Ling-Ning Li, Enable collaborative learning: an improved e-learning social network exploiting approach, Proceedings of the 6th Conference on WSEAS International Conference on Applied Computer Science, Vol.6, 2007, pp.311-314.

## Appendix A:

Selected Top20 Mashups and Patterns (Due to the limit of space, we only list 20 of them)			
Mashup	APIs	Notes	Tags
Wii Seeker	Amazon eCommerce+eBay+ Google Maps	A Mashup to help consumers locate a Nintendo Wii. Provides retail addresses, locations, shipment dates, and local ebay auctions.	auction, mapping, shopping
Flash Earth	Google Maps+Microsoft Virtual Earth+ NASA+OpenLayers+ Yahoo Maps	Zoomable Mashup of Google Maps, Virtual Earth and other satellite imagery through a Flash application. Try rotating the compass or building a permanent link to a location.	mapping
Celebrity Sexy Video Finder	Google Ajax Search+ YouTube	Not safe for work, NSFW. Search and find all sexy YouTube and Google videos of your favorite female and male celebrities. Contains adult content.	celebrity, movies, search, sex, video
Weather Bonk	Google AdWords+ Google Maps+ hostip.info+ Microsoft Virtual Earth+ NASA+ NOAA Weather Service+ WeatherBug+Yahoo Geocoding+ Yahoo Maps+ Yahoo Traffic	Rich Mashup with live weather, forecasts, webcams, and more on a Google Map.	mapping, weather, webcams
Beam Me Up, Hottie	HotOrNot	Uses the HotOrNot API to find girls and guys in your by location	dating, mapping, search
Google Maps Flight Simulator	Google Maps	Goggle is a very fun and creative flight simulator built on top of Google Maps. Pick from one of five cities and go. Change altitude and direction using the arrow keys.	games, mapping
LivePlasma	Amazon eCommerce	A visually rich application that combines the Amazon API to show the relationship between movies, bands, actors, etc. You can go straight from interacting to making purchases.	search, shopping
1001 Secret Fishing Holes	Google Maps	Over a thousand fishing spots in national parks, wildlife refuges, lakes, campgrounds, historic trails etc.	mapping, sports
BidNearBy	eBay+ Google Maps	Search local auctions and classified listings (craigslist) and see where they are located on a map view.	auction, craigslist, mapping, shopping
Where's Tim Hibbard?	Google Maps	See where Tim is now: he always carries a GPS-enabled mobile phone and data gets plotted on a Google Map.	gps, mapping
25 Unsafest US Cities	GoogleMaps	A look at the 25 most unsafe cities in America.	crime, mapping, trivia
Sad Statements	YahooTerms+Twitter+ Flickr	Sad tweets are grabbed from Twitter and illustrated via pictures from Flickr.	blog, messaging, microblogging, photo, ruby, widgets
Goocam	GoogleMaps	A Google map of unprotected/open camera streams obtained from Google searches. The IP addresses for each camera's url have been mapped to its Geo-location. Sometimes loads slowly. Some cams NSFW.	mapping, webcams
Famous London Barristers	GoogleMaps	View the 10 most famous London barristers, as listed by thelawyer.com.	law, london, mapping, uk
Celebrity Photos Gallery	Yahoo+YahooImages	A dyanamic collection of celebrity photo galleries. Data comes from Yahoo Image Search and Yahoo Search.	celebrity, movies, photo, television
BBC News Map	GoogleMaps+BBC	See where the latest news is happening in the UK.	mapping, news, uk
FlickrSudoku	Flickr	Mashup of a popular online sudoku player and Flickr. Sudoku player courtesy sudoku.com.au.	games, photo
Adactio Elsewhere	Amazon+del.icio.us+Flic kr+Upcoming	J. Keith combines a variety of personal information from across the web in one place via Ajax scripting.	bookmarks, events, photo, shopping
Twitter Top News Trends	YahooTerms+Twitter+Go ogle+GoogleCustomSearc h+AmazonS3	Top terms often appear on Google Trends before the story people are searching for gets indexed. Googling these terms often leads to stale news. TTNT attempts to explain breaking news by gathering relevant information from Twitter.	news, search, Twitter
50 Top US Medical Schools	GoogleMaps	Map of the top US medical schools are reported by US News. Pop-up markers show MCAT scores and average tuition.	education, mapping