Making Mashups with Marmite: Towards End-user Programming for the Web

Jeffrey Wong & Jason I. Hong

Human-Computer Interaction Institute
Carnegie Mellon University
The Problem

Lot of information on the web

But not always connected in ways which are useful.

One Solution: Mashups

Websites that combine functionality and content from different websites

Example: housingmaps.com
$3200 4 Bed/3 Bath House for Rent (cupertino)

$1750/3br - Town house for rent (san jose east)

$2250/3br - 2Bath Home near Westfield Valley Fair (san jose west) pic

$2000/4br - 4RM/2BD Spacious Quiet Single Family House (san jose south) pic

$1400/2br - 2BR 1B in Watsonville $1,400/mo (watsonville)

$875/1br - Adorable apartment in a very peaceful neighborhood (santa clara)

$850 ADORABLE Cottage Studio in Ben Lomond!!! (Ben Lomond)

$2600/3br - Great Location (capitola)

$950/1br - 1 bd apt (campbell)
San Jose, CA

Google Maps
Craigslist apartment listings + Google Maps = Housingmaps.com
Lots of Mashups

According to programmableweb.com:

• 1019 mashups in Sept 2006
• 1861 mashups as of this morning
  (up from 1852 yesterday)
• 3.14 new mashups per day
  \(\pi\) new Mashups per Day!
Mashups are HARD

Requires lots of programming expertise:

Text parsing

Pattern Matching

Web Services APIs

Databases

http://www.google.com/calendar/feeds/default/private/full?start-min=2006-03-16T00:00:00&start-max=2006-03-24T23:59:59
# Web EUP Systems

## Programming Languages
- **Chickenfoot**  
  (Bolin et al., 2006)
- **Greasemonkey**

## Content Selection
- **Creo and Miro**  
  (Faaborg & Lieberman, 2006)
- **Sifter**  
  (Huynh et al., 2006)
- **Dapper**  
  **KOALA**  
  (Little et al., 2007)

## Mashup Platforms
- **Yahoo! Pipes**
- **OpenKapow**
- **QEDWiki**

## Webpage Integration
- **C3W**  
  (Fujima et al., 2007)
- **Hunter/Gatherer**  
  (m.c. schraefel et al., 2002)
- **Internet Scrapbook**  
  (Sugiura et al., 1998)
Our Solution: Marmite

Lets user create a data-flow to process web pages

1. Extract listings of events.
2. Filter out things I can’t go to.
3. Visualize events on a calendar

Make it easy to create web mashups:

Without learning a programming language

Without

Runs in your web browser
How Marmite is different

- Access to Web Service APIs
- Combine Web Service APIs with screen-scrape oriented programming
- Data flow view and data view
Talk Outline

Problem Statement

Formative Study

Prototype Overview

Evaluation
Formative Study
Initial Apple Automator study

Operators

Data-flow

Find Messages in Mail
- Find: Messages
- Whose: Message Contents Contains apple

Combine Mail Messages
- Mail messages

Text to Audio File
- System Voice: Victoria
- Save As: Apple Mail Audio
- Where: Desktop

Import Audio File
- Using: AAC Encoder
- Delete source files after encoding

Library
- Applications
  - Address Book
  - Automator
  - DVD Player
- Finder
  - Font Book
  - iCal
  - Image Capture
  - iTunes
  - Mail
  - PDF
  - Preview
  - QuickTime Player
  - Safari
  - Spotlight
  - System
  - TextEdit
- New Text File

Action
- Ask For Servers
- Connect to Servers
- Create Archive
- Find Finder Items
- Get Folder Contents
- Get Specified Servers
- Launch Application

Options

Workflow Execution Failed
My new alarm clock
Automator Results

1. No feedback between operations
   - Intermediate feedback or incremental development

2. Network slows development
   - Support operating on samples of data

3. Selecting operations is hard
   - Provide context-specific suggestions
Talk Outline

Problem Statement

Formative Study

Prototype Overview

Evaluation
### Step 2: Filter Events

1. **Find events (EVDB)**

   - **Inputs**
     - Get: from column:
     - Date: in Start Time
   - Remove events...
   - Refine:
   - **Outputs**
     - Displays: View results for this step

2. **Filter Events**

   - **Inputs**
     - Get: from column:
     - Date: in Start Time
   - Remove events...
   - Refine:
   - **Outputs**
     - Displays: View results for this step

---

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyber Senior</td>
<td>2007-04-20</td>
<td>Lynn Williams</td>
<td>Pitts...</td>
<td>40.48...</td>
<td>-80.03...</td>
</tr>
<tr>
<td>Cyber Senior</td>
<td>2007-04-20</td>
<td>Lynn Williams</td>
<td>Pitts...</td>
<td>40.48...</td>
<td>-80.03...</td>
</tr>
<tr>
<td>Cyber Senior</td>
<td>2007-04-20</td>
<td>Lynn Williams</td>
<td>Pitts...</td>
<td>40.48...</td>
<td>-80.03...</td>
</tr>
<tr>
<td>Friends of th...</td>
<td>2007-04-20</td>
<td>Green Tree F...</td>
<td>Pitts...</td>
<td>40.44...</td>
<td>-79.98...</td>
</tr>
<tr>
<td>Cyber Senior</td>
<td>2007-04-20</td>
<td>Lynn Williams</td>
<td>Pitts...</td>
<td>40.48...</td>
<td>-80.03...</td>
</tr>
<tr>
<td>Cyber Senior</td>
<td>2007-04-20</td>
<td>Lynn Williams</td>
<td>Pitts...</td>
<td>40.48...</td>
<td>-80.03...</td>
</tr>
<tr>
<td>Cyber Senior</td>
<td>2007-04-20</td>
<td>Lloyd Mcbride...</td>
<td>Pitts...</td>
<td>40.46...</td>
<td>-79.90...</td>
</tr>
<tr>
<td>Cyber Senior</td>
<td>2007-04-20</td>
<td>Lloyd Mcbride...</td>
<td>Pitts...</td>
<td>40.46...</td>
<td>-79.90...</td>
</tr>
<tr>
<td>Distinctive D...</td>
<td>2006-10-14</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Distinctive D...</td>
<td>2006-10-14</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Distinctive D...</td>
<td>2006-10-14</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Distinctive D...</td>
<td>2006-10-14</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Distinctive D...</td>
<td>2006-10-14</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Distinctive D...</td>
<td>2007-01-18</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Gritty Bits: N...</td>
<td>2007-01-20</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Gritty Bits: N...</td>
<td>2007-01-20</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Backyard Inv...</td>
<td>2007-02-06</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Bizarre Beasts</td>
<td>2007-02-19</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Mezzotint in...</td>
<td>2007-03-03</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
<tr>
<td>Marshes: The...</td>
<td>2007-03-03</td>
<td>Pitts...</td>
<td>40.42...</td>
<td>-79.98...</td>
<td></td>
</tr>
</tbody>
</table>

---

Next operator suggestions.

Replace this placeholder by selecting the next operator from operator list or the suggestions list.

Show suggestions
Operators
Show: All

Workflow

Dragging operators here...

Results from operators will appear here.
Adding an operator to scrape web page
Scraping links off a web page by demonstration
Extracting the address from the web page at each URL
Geocoding each address, reconciling input type...

Step 2: Extract Address

![Workflow diagram showing steps 1 and 2: Select Links From Page and Extract Address.]

Next operator suggestions:
Replace this placeholder by selecting the next operator from the operator list or the suggestions list.
Show suggestions
Loading each address into Yahoo! maps

**Workflow**

1. Select Links From Page
   - ![Select Links From Page](image)
2. Extract Address
   - ![Extract Address](image)
3. Geocode
   - ![Geocode](image)

**Step 3: Geocode**

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2000</td>
<td>3br -</td>
<td><a href="http://sfbay...7159">http://sfbay...7159</a> Rouse C... San+Jose</td>
<td>Jose</td>
<td>$1726</td>
<td>2br -</td>
<td><a href="http://sfbay...100">http://sfbay...100</a> buckingh... SANTA+CL + CLARA</td>
</tr>
</tbody>
</table>
Operators

input types

output types

Extract Address

Get: from column:
Web Page: B: URL

Outputs

Write: to column:
Street Address: new column
City: new column
State: new column
Displays: View results for this step
Operator Types

- sources
- filters
- processors
- sinks
Design solutions

Problem #1:

little feedback about state of system between operations

Solution: link data flow and data view together

Use hybrid data flow / data view, showing an operation and its effects together

Data views can be:

- tables (usually)
- maps
Design solutions

Problem #2: difficult to iterate due to network speeds

Solution: *provide fine grained control*

Reload, Pause, & Play
Problem #3: Selecting starting and next operators

Solution:

Suggest next actions

Use data types to select next operations

Filter operators to only show relevant ones
Step 4: Yahoo Maps!

3. Geocode

2. Extract Address

Next operator suggestions.

Replace this placeholder by selection the next operator from operator list or the suggestions list.

Show suggestions
Talk Outline

Problem Statement

Formative Study

Prototype Overview

Evaluation
Think-Aloud Evaluation

Informal user study with 6 people

- 2 novices
- 2 people with spreadsheet experience (formulas)
- 2 people with programming experience

4 Tasks (in increasing difficulty)

- Warmup task showing how to retrieve a set of addresses and how to geocode an address
- Search for and filter out events further than a week away
- Compile a list of events from two event services and plot them on a map
- Recreate the map from housingmaps website
Results

3 people able to complete all 4 tasks in ~1 hour (1 spreadsheet, 2 programmers)

• recreate housingmaps map in 15 minutes

• wished to be contacted when Marmite was ready

First two users (novices) confused about suggested actions (automatically popped up, made manual for other 4 users)

Novices made some progress, not able to finish all tasks
Discussion

Biggest problem was understanding the data flow

- Did not understand input and output concept
- Applied operators as one-off
- Did not understand data flow and data view were linked

Could have been due to visual design, reuse of results panel

One-off operations may be important to support
Future Work

Support

• saving

• parameterization

Semi-automatic generation of operators

Work with real-time data-feeds
• Platform for mixing web services without programming

• Data-flow programming environment with incremental feedback

• Integrate screen-scraping with visual programming approach to EUP

Thanks!

Jeff Wong
jeffwong@cmu.edu

Generously funded by:
NSF SGER IIS-0646526
DARPA
Microsoft Research SenseWeb
Lo-fidelity prototypes

6 paper prototypes with 20 participants
End-User Programming (EUP)

A Small Matter of Programming
Bonnie Nardi (1993)

Watch What I Do
Allen Cypher (1993)

Your Wish Is My Command
Henry Lieberman (2001)

End User Development
Lieberman et al. (2006)
EUP for Web is Different

“Screen-scraping” to get data

Features never designed to work together (even in APIs)

Different formats, types, and semantics

Lots of data

Many operations go over the network

Access restrictions (keys and authentication)