Google Ajax Search API

CSE 190 M (Web Programming), Spring 2007
University of Washington

References: <u>homepage</u>, <u>code samples</u>, <u>API reference</u>

Javascript frameworks

Many groups have released Javascript frameworks you can include in your web pages to add effects:

- <u>Yahoo! UI</u> : set of rich UI widgets (demos <u>1</u>, <u>2</u>, <u>3</u>)
- <u>Scriptaculous</u> : rich UI animation/effects library
- <u>Prototype</u> : DOM/Ajax extender
- <u>jQuery</u> : DOM/event library
- <u>Dojo</u> : graphics (<u>demos</u>)
- <u>Sarissa</u> : XML-related tools
- Google <u>Ajax Search API</u>, <u>Ajax Feeds API</u>, <u>Maps API</u>, <u>Web Toolkit</u>

What is the Google Ajax Web API?

- a set of Javascript objects that allow you to perform Google searches
  - web, local, map, image, video, blog, news, books
- you can manipulate the search results or place them on your web page
- implemented as a `.js` file you include in your web page

API keys

- the API requires a "key" that is bound to your web domain (such as `www.cs.washington.edu` or `students.washington.edu`)
- you can <u>sign up</u> to get a key for a particular domain
- code to use API on `https://students.washington.edu`:

```
<script src="http://www.google.com/uds/api?file=uds.js&amp;v=1.0&amp;key=
ABQIAAAAjwpa3Lmr7L0oXrBemAIp0BSx0OSUw7R9ozDoChrVVXrbgKdJeBTWWRQkTwM5gxzSfgnTYruTJLircA" type="text/javascript"></script>
```

- (must remove line breaks from above code)

Ways of using the API

- onscreen search controls
  - simple
  - not much customization in appearance or behavior
- raw search results
  - more work
  - very customizable

"Hello, World!" app

```
// in your window.onload handler:
var searchControl = new GSearchControl();
searchControl.addSearcher(new search());
searchControl.draw(document.getElementById("id"));
searchControl.execute("query");
```

- id is the CSS ID of a div where you want the search to show up
- search is one of the following: GwebSearch, GlocalSearch, GvideoSearch, GblogSearch, GnewsSearch, GbookSearch
  - can add multiple searches to the same GSearchControl
- query is what you initially want to search for (optional)
- complete HTML file: for www.cs.washington.edu, for students.washington.edu

Search control example

hamburger   Search
powered by Google™

Local
(3)

Hamburger Mary's
1525 E Olive Way
Seattle,
WA
USA
(206) 324-8112
directions
Get directions:
To here
-
From here
clipped from Google - 5/2007
Web
(5)

Hamburger - Wikipedia, the free encyclopedia
A **hamburger** is a sandwich that consists of a cooked patty of ground meat that is
fried, steamed, grilled, or broiled, and is generally served with various **...**
en.wikipedia.org
en.wikipedia.org/wiki/Hamburger
clipped from Google - 5/2007

Video
(4)

Backwards Hamburger
Our FAST FOOD NATION exposed. Before you take another bite, you must see this movie
(there's shit in the meat!) Take action at Backwardshamburger.co...
Nov 13, 2006
www.youtube.com
clipped from Google - 5/2007
Blog
(4)

Hamburger Hit
May 07, 2007
3 hours ago
Haniya and Jeff requested toasted buns and the rest of us had plain buns. I spread Vegenaise and homemade
ketchup on each bun. I topped the **hamburger** patties with raw goat cheddar cheese and onion slices sauteed in
red palm oil. **...**
http://suchtreasures.wordpress.com
clipped from Google - 5/2007

The search objects

search object	analogous to
GwebSearch	www.google.com
GlocalSearch	local.google.com
GvideoSearch	video.google.com
GblogSearch	?
GnewsSearch	news.google.com
GbookSearch	?

Methods/properties common to all searches

- `new search()`
- `execute("query")`
  tells the search control to search for the given text
- `setSearchCompleteCallback(object, method)`
  tells the search control to call the given method on the given object when the search is completed
- `results[]`
  an array of all search results found

Web search example

```
var search = new Search();
search.setSearchCompleteCallback(window, function() {
    for (var i = 0; i < search.results.length; i++) {
        do something with search.results[i];
    }
});
search.execute("query");

var search = new GwebSearch();
search.setSearchCompleteCallback(window, function() {
    for (var i = 0; i < search.results.length; i++) {
        alert(search.results[i].url);
    }
});
search.execute("miserable failure");
```

Click me

Web search: `GwebSearch`

- elements of `results[]` have the following properties:
    - `url`
    - `content` : a snippet of the page's text
    - `titleNoFormatting` : search result title

Local search: `GlocalSearch`

- elements of `results[]` have the following properties:
    - `url`
    - `titleNoFormatting` : search result title
    - `lat`, `lng` : latitude/longitude coordinates
    - `streetAddress`, `city`, `region`, `country`
    - `phoneNumbers[]`
    - `ddUrl`, `ddUrlToHere`, `ddUrlFromHere` : driving directions URLs
- `GlocalSearch` method: `setCenterPoint("location")`
    - `search.setCenterPoint("Seattle, WA");`
    - `search.setCenterPoint("98105");`

Video search: `GvideoSearch`

- eleuents of `results[]` have the following properties:
    - `url`
    - `content` : a snippet of the video's text
    - `titleNoFormatting` : search result title

- `duration` : length of clip in seconds
- `tbUrl`, `tbWidth`, `tbHeight` : thumbnail image info
- `playUrl` : address of Flash video that can be embedded

Blog search: `GblogSearch`

- elements of `results[]` have the following properties:
  - `postUrl`, `blogUrl` : web address of this post and entire blog
  - `content` : a snippet of the blog post's text
  - `titleNoFormatting` : search result title
  - `author`, `publishedDate`

News search: `GnewsSearch`

- elements of `results[]` have the following properties:
  - `url`
  - `content` : a snippet of the blog post's text
  - `titleNoFormatting` : search result title
  - `location` (e.g. "Edinburgh,Scotland,UK")
  - `publishedDate`
  - `relatedStories[]`

Book search: `GbookSearch`

- elements of `results[]` have the following properties:
  - `url`
  - `content` : a snippet of the blog post's text
  - `titleNoFormatting` : search result title
  - `authors`
  - `publishedYear`
  - `pageCount`

Practice problem: Google Fight

- Write a page similar to Google Fight, where two Google search queries compete against each other, except that your page should determine the winner of the "fight" by the query that has the longest URL in any of its search results.