www.archive.org
415.561.6767
415.840-0391 e-fax

Internet Archive
300 Funston Avenue
San Francisco, CA 94118

AFFIDAVIT OF DUNCAN HALL

1. I am a Records Request Processor at the Internet Archive, located in San Francisco, California. I make this declaration of my own personal knowledge.

2. The Internet Archive is a website that provides access to a digital library of Internet sites and other cultural artifacts in digital form. Like a paper library, we provide free access to researchers, historians, scholars, and the general public. The Internet Archive has partnered with and receives support from various institutions, including the Library of Congress.

3. The Internet Archive has created a service known as the Wayback Machine. The Wayback Machine makes it possible to browse more than 450 billion pages stored in the Internet Archive's web archive. Visitors to the Wayback Machine can search archives by URL (i.e., a website address). If archived records for a URL are available, the visitor will be presented with a display of available dates. The visitor may select one of those dates, and begin browsing an archived version of the Web. Links on archived files in the Wayback Machine point to other archived files (whether HTML pages or other file types), if any are found for the URL indicated by a given link. For instance, the Wayback Machine is designed such that when a visitor clicks on a hyperlink on an archived page that points to another URL, the visitor will be served the archived file found for the hyperlink's URL with the closest available date to the initial file containing the hyperlink.

4. The archived data made viewable and browseable by the Wayback Machine is obtained by use of web archiving software that automatically stores copies of files available via the Internet, each file preserved as it existed at a particular point in time.

5. The Internet Archive assigns a URL on its site to the archived files in the format http://web.archive.org/web/[Year in yyyy][Month in mm][Day in dd][Time code in hh:mm:ss]/[Archived URL] aka an "extended URL". Thus, the extended URL http://web.archive.org/web/19970126045828/http://www.archive.org/ would be the URL for the record of the Internet Archive home page HTML file (http://www.archive.org/) archived on January 26, 1997 at 4:58 a.m. and 28 seconds (1997/01/26 at 04:58:28). The date indicated by an extended URL applies to a preserved instance of a file for a given URL, but not necessarily to any other files linked therein. Thus, in the case of a page constituted by a primary HTML file and other separate files (e.g., files with images, audio, multimedia, design elements, or other embedded content) linked within that primary HTML file, the primary HTML file and the other files will each have their own respective extended URLs and may not have been archived on the same dates.

6. Attached hereto as Exhibit A are true and accurate copies of screenshots of the Internet Archive's records of the archived files for the URLs and the dates specified in the attached coversheet of each printout.

7. I declare under penalty of perjury that the foregoing is true and correct.

DATE: 07/20/2021

Duncan Hall

Duncan Hall

EXHIBIT A

https://web.archive.org/web/20060705114622/http://www.google.com/apis/reference.html

Google

Google SOAP Search API (beta)

Google SOAP Search API Reference

Contents

1. Overview

Back to top

This document explains in detail the semantics of the function calls you can make using the Google SOAP Search API service. In this document, you will learn:

- How Google's query syntax works.
- How to restrict your queries to portions of Google's index, such as a particular language or country.
- How to interpret the search results information sent back by the Google SOAP Search API service.

You may also find the following files from the Google SOAP Search API developer kit to be helpful:

- GoogleSearch.wsdl - WSDL description for Google SOAP Search API SOAP interface.
- soap-samples/ - example SOAP messages and responses.
- javadoc/index.html - javadoc for the example Java libraries.

This is a beta document. If you have comments, find errors, or just have questions, please use the Google SOAP Search API discussion forum.

1.1 Search Requests

Back to top

Search requests submit a query string and a set of parameters to the Google SOAP Search API service and receive in return a set of search results. Search results are derived from Google's index of billions of web pages.

The details of the interactions involved with search requests are covered in the Search Request Format and Search Results Format sections of this document.

1.2 Cache Requests

Back to top

Cache requests submit a URL to the Google SOAP Search API service and receive in return the contents of the URL when Google's crawlers last visited the page (if available).

Please note that Google is not affiliated with the authors of cached pages nor responsible for their content.

The return type for cached pages is base64 encoded text.

1.3 Spelling Requests

Back to top

Spelling requests submit a query to the Google SOAP Search API service and receive in return a suggested spell correction for the query (if available). Spell corrections mimic the same behavior as found on Google's Web site.

Spelling requests are subject to the same query string limitations as any other search request. (The input string is limited to 2048 bytes and 10 individual words.)

The return type for spelling requests is a text string.

2. Search Request Format

Back to top

2.1 Search Parameters

Back to top

This table lists all the valid name-value pairs that can be used in a search request and describes how these parameters will modify the search results.

Name	Description
key	Provided by Google, this is required for you to access the Google service. Google uses the key for authentication and logging.
q	(See Query Terms section for details on query syntax.)
start	Zero-based index of the first desired result.
maxResults	Number of results desired per query. The maximum value per query is 10. **Note:** If you do a query that doesn't

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.