



Searching PDF - 2019

Leonard Rosenthol | Senior Principal Architect, PDF & DocCloud



- How hard can it be to search content?
 - It's easy
 - Everybody does it
 - Lots of “off the shelf” solution



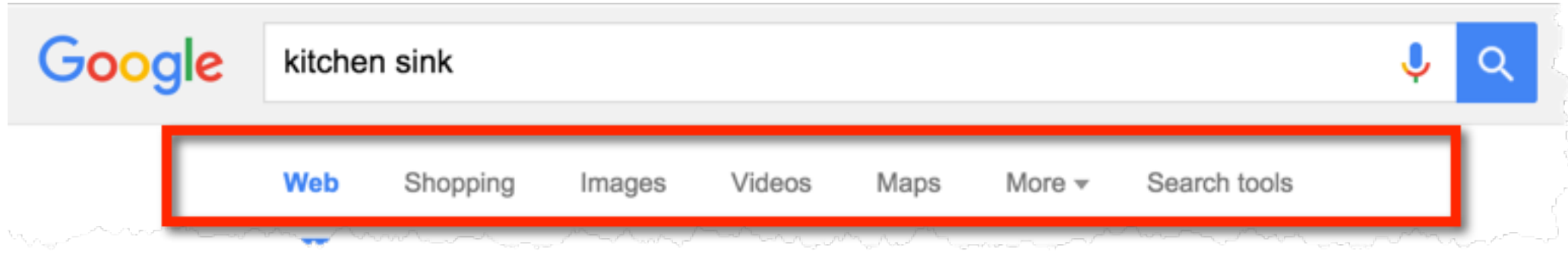
Indexing: Behinds the scenes of Search

- A storage system for the collected information that is organized (and optimized) for the fast and accurate retrieval of information via a query.
 - w/o an index, you'd have to scan/process every document, every time!
- Size vs. Speed
 - How much do you store?
 - How fast do you want to find something?
 - How easy is it to update and maintain?

Schema or not?

- What (and how) do you store in the index?
- Most common approach is a schema where all the data in a single format or structure
 - This makes it really easy to find things as it's well organized
 - Useful for common data sets – data from form-like workflows
- Alternative is a more “free form” approach using a structured language (XML or JSON)
 - Elasticsearch – most popular open-source search engine works this way
- Or find a common, structured, format that is easy to work with...HTML

What are you searching & indexing?



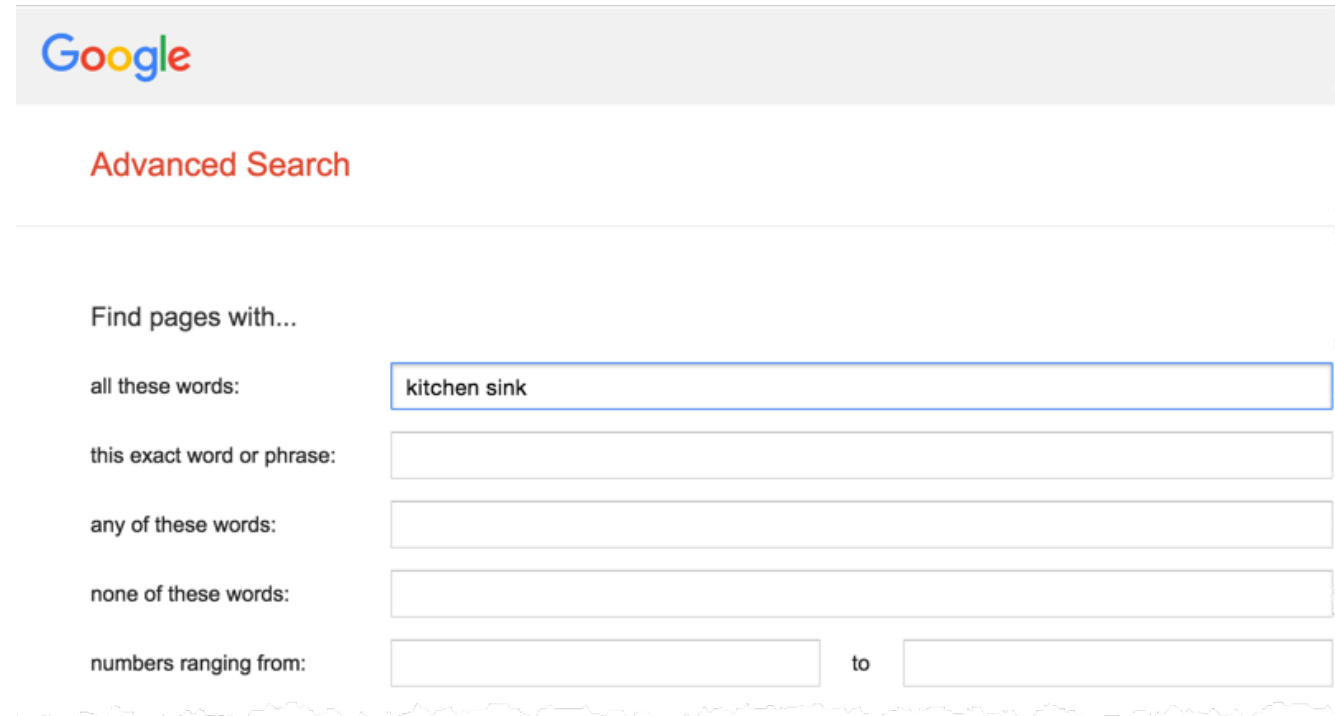
- Notice how Google shows you different types of things you can search?
- Each one is indexed separately using a different approach that is optimized for the content in question

But what about documents/content?

- Can you have a common index for various document formats?
- Does the index for a spreadsheet resemble that of a presentation or long form document?
- How do you handle paginated docs vs. those that are not (eg. HTML)?
- Lowest Common Denominator approach == plain text
 - This is the approach used by most, since they only think about the text...
<https://www.elastic.co/guide/en/elasticsearch/plugins/5.6/ingest-attachment.html>
- Lower Common Denominator approach == HTML
 - this is what many of the online search engines (eg. Google) do (who also want images, etc.)

Queries

- Single word: “sink”
- Simple text/phrase: “kitchen sink”
- Booleans: “kitchen or bathroom sink”
- Proximity
 - Does it have to be “kitchen sink” or would “sink in kitchen” also match?



The image shows a screenshot of the Google Advanced Search interface. At the top is the Google logo. Below it, the text "Advanced Search" is displayed in red. Underneath, the heading "Find pages with..." is followed by several search criteria options, each with a corresponding text input field:

- "all these words:" with an input field containing the text "kitchen sink".
- "this exact word or phrase:" with an empty input field.
- "any of these words:" with an empty input field.
- "none of these words:" with an empty input field.
- "numbers ranging from:" with two empty input fields separated by the word "to".

- Filtering
 - Other criteria that can be applied to the query to help determine if a given match is the correct one

Then narrow your results by...

language:	<div>any language</div>
region:	<div>any region</div>
last update:	<div>anytime</div>
site or domain:	<div></div>
terms appearing:	<div>anywhere in the page</div>
SafeSearch:	<div>Show most relevant results</div>
file type:	<div>any format</div>
usage rights:	<div>not filtered by license</div>

Searching filenames in Acrobat

query:

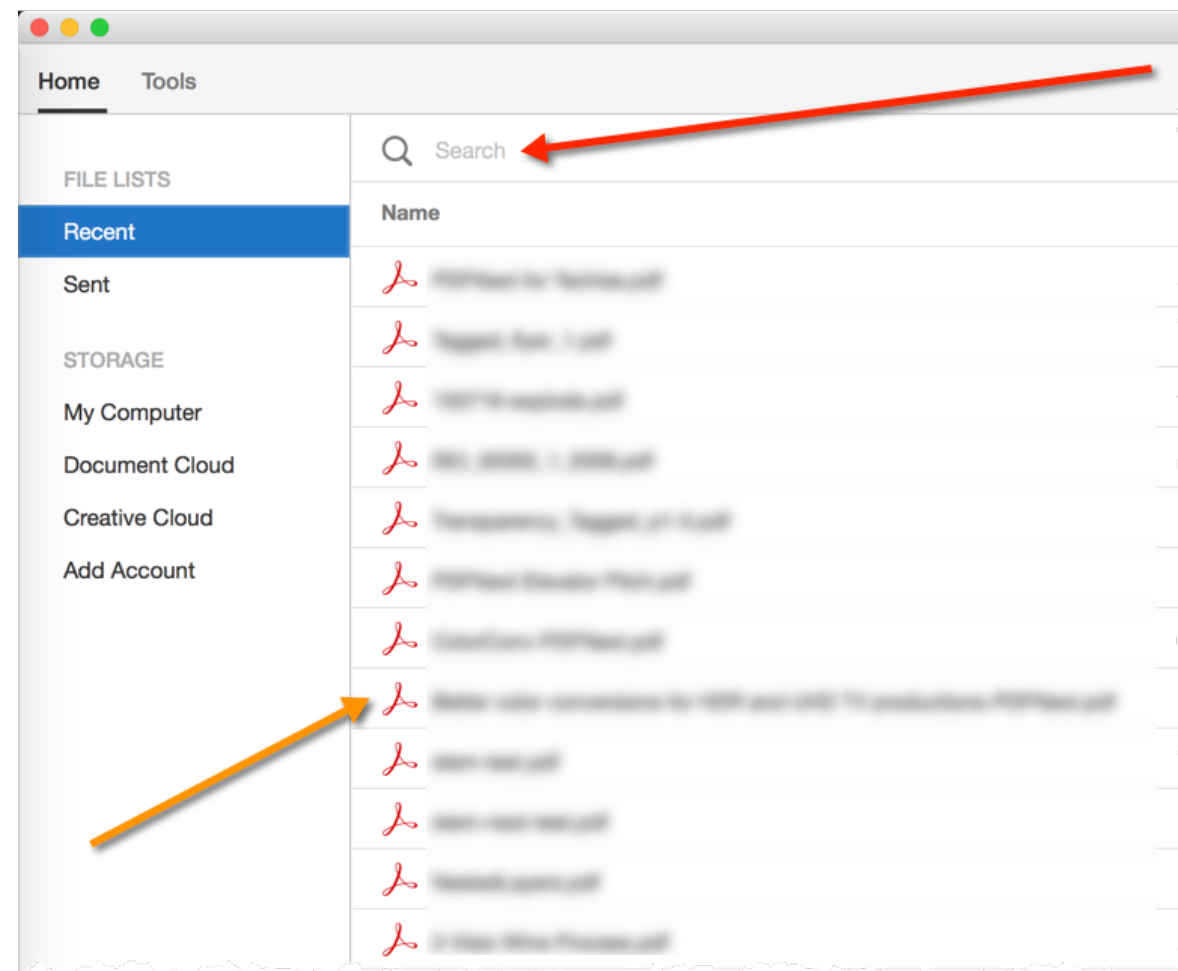
(name:sink*) AND

((ext:pdf) OR

(content_type:application/pdf))

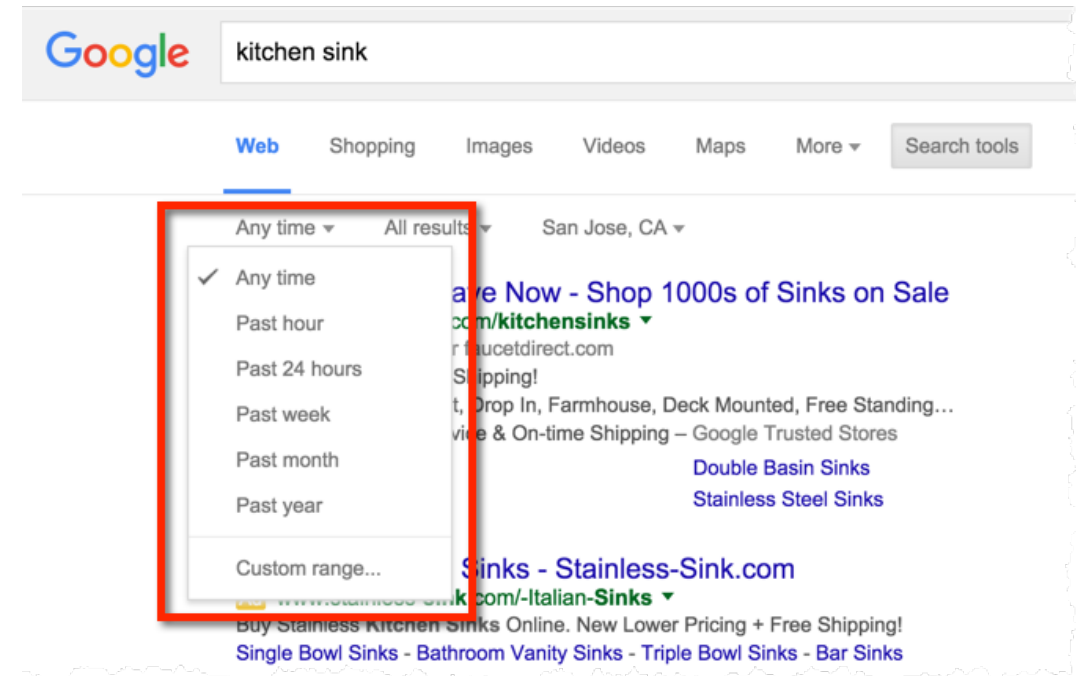
sort: name asc,created desc

return-fields: : ["name", "size",
"content_type", "created"]



Relevance and Ranking

- How do you determine what is the most relevant results from the query?
- Sorting
 - Ordering criteria, such as date
- Google Page Rank
 - Application of social graph aspects to the relevancy & ranking criteria
 - “Things not Strings”
- Semantic Understanding
 - If you understood the true semantics of a document vs. just their content you could rank results. (eg. Headings vs. body)



Content Search is NOT easy – because PDF is complex



ISO 32000-2
974 Pages

Content Search is NOT easy: What are you searching?

- Documents are not just about “plain text”
 - Rich Semantics
 - Simple (heading, table) & Structural (chapter, story, article)
 - Publishing Elements
 - TOC, index, glossary, foot/endnotes
 - Raster and Vector Images
 - Media (audio, video, 3D)



Can a single index contain it all? YES – if you plan ahead!

Pages vs. Pages

- In order to find where something came from, you also store a reference to the source in the index along with its data
 - Each of these is commonly called a “document”
 - And for HTML – a “page” == a “document”. Nice!
- But what to do you do with PDF, with multiple pages?
 - One “document” per “document”?
 - One “document” per “page”?
- You want to know which page(s) the results are on – so you might think per-page...
 - But what about a cross-page search? How would that work? Right – it doesn’t!
 - (or at least it doesn’t rank well)



Common
Mistake

- Structure and Tagging were added to PDF in 1.4
 - Unfortunately only about 15% of all PDFs actually use it
 - PDF/UA becoming mandated will help this!
 - Many tools exist for converting a PDF to tagged PDF
 - Or even just extracting out the semantics of the PDF with the content
- Content extraction w/o semantics yields poor results
 - Not to mention lack of understanding of reading order!

A yellow starburst graphic with a black outline, containing the text "Common Mistake".

Common
Mistake

Metadata Search – also NOT easy

- ANY metadata is better than no metadata! (so starting with DocInfo is good...)
 - **Not everyone bothers to do even this (eg. Google!)**
- XMP's richness and extensibility is a blessing and a curse
 - Customers use custom schemas – but w/o any documentation on those schemas
 - Need to be able to understand them to offer intelligent search
 - XML vs. JSON (ISO 16683-4 will provide a standard serialization to JSON-LD)
- Embedded images & media
 - Need to be able to extract, index and manage as a component of the document
 - “Find all images with my copyright” needs to find them in documents too
 - And know page/location & list of occurrences (if used multiple times in the same doc)



Searching Attachments, Annotations and more...

- Annotations & Markup Search
 - Need to be able to extract, index and manage as a component of the document
 - “Find that comment I wrote to Bill about his crazy project idea last year”
- Attachment Search
 - How do you index and catalog a document inside another document?
 - Embedded Files in a PDF
 - Attachments in an Email
 - Contents of a ZIP

Nobody does this

<https://lifehacker.com/how-can-i-search-through-pdf-comments-1833299295>



Content Search is NOT Easy: At Cloud Scale

- Performance
 - how quickly can a new doc be added to the index, to avoid user confusion?
- Scalability
 - How many pages (from how many documents)
- Eventual consistency
- Overall robustness/reliability/availability



Questions





Adobe

Searching Forms

- Forms are a common type of document, where the form's data is not necessarily the content
 - Radio Buttons & Checkboxes
 - List Boxes (with one or more selection)
 - Comb fields
- Need to understand not only the values but the semantics of the field to which that value belong
 - Not necessarily the name of the field (though that's useful too)
 - Content such as “List of elements” (eg. items on an invoice)
- Oh – and not every “form” is already structured as one...



Application form

Please fill in ALL the boxes below in BLOCK CAPITALS, using black ink.
If you miss something out it might delay your application.
Remember to read the Agreement conditions booklets enclosed and the

Step 1 – Your personal details

Title Surname Middle name

First name Date of birth

Are you? Male ☐ Female ☐

Nationality

Mother's maiden name

Home phone (include dialling code)

Content Search is NOT Easy: Enterprise Considerations

- Compliance – PCI, SOC2, HIPPA, etc.
- Data Sovereignty
- ACL's & Permissions



- Solvable problems provided they are incorporated from the start

Having access to this data means more than just search

- Document Classification
- Document Summarization
- Document Recommendation
 - locating similar documents
- Smart Form Filling
- And many other things...

