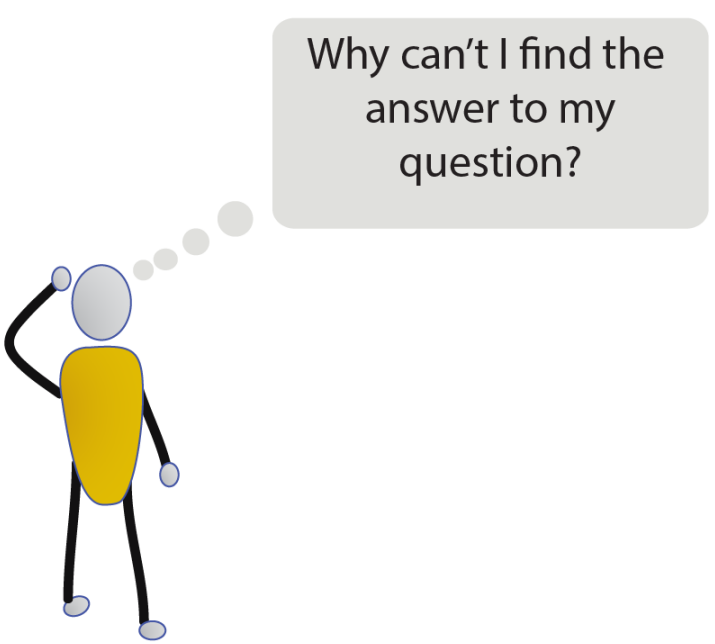
Make Your Content More Findable When Users Browse and Search

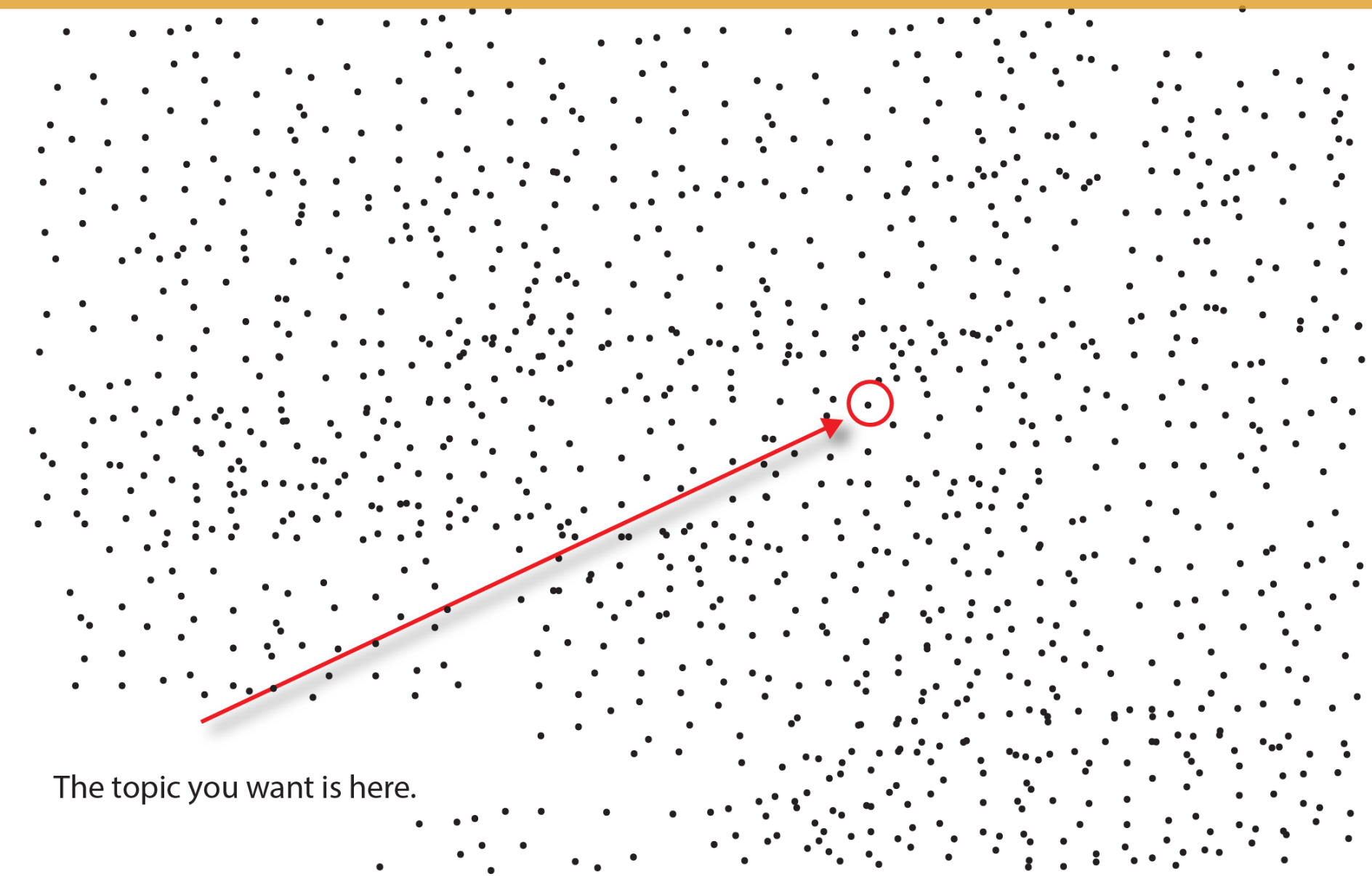
By Tom Johnson

UA Europe June 2013

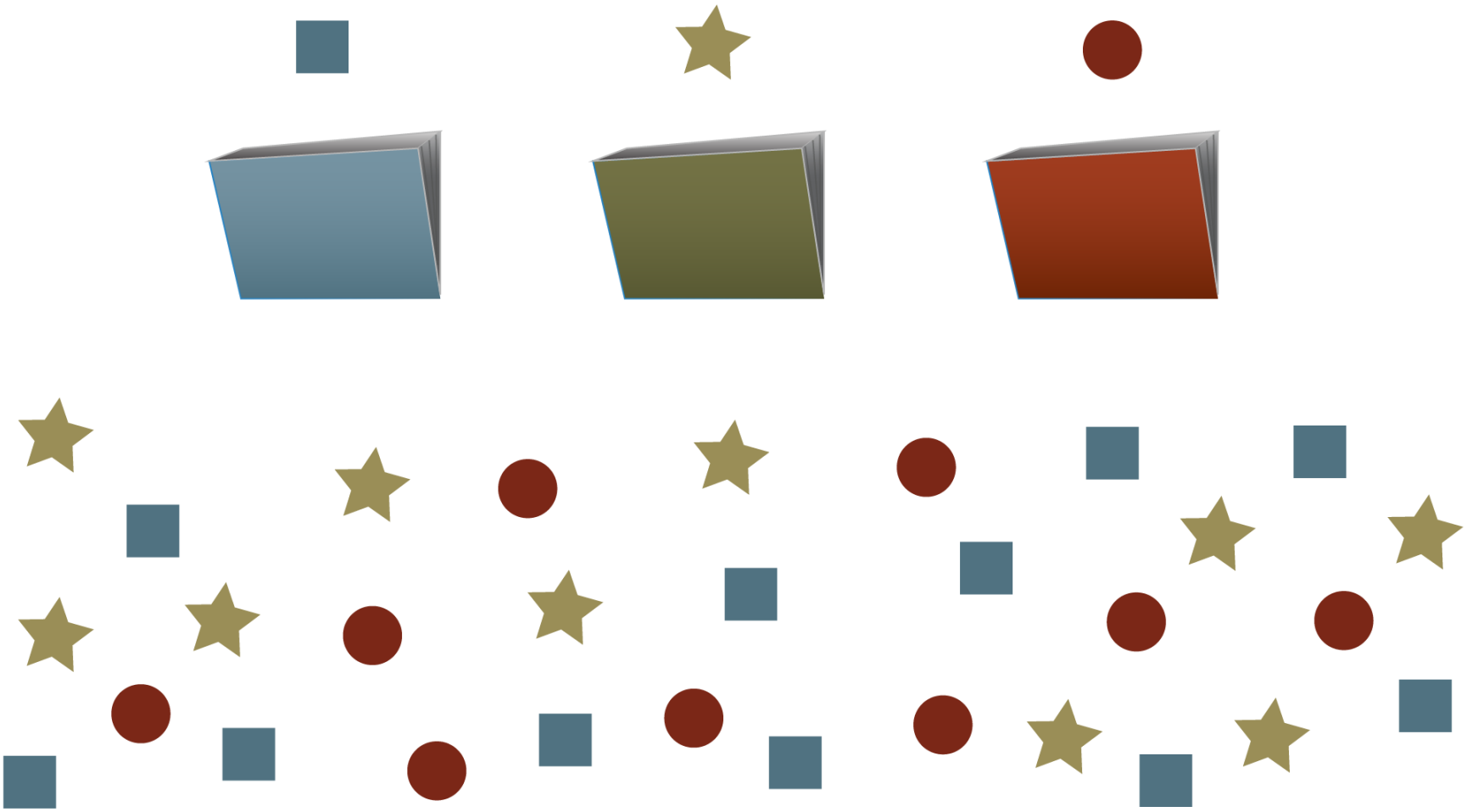


Tom Johnson
@tomjohnson, #techcomm
idratherbewriting.com

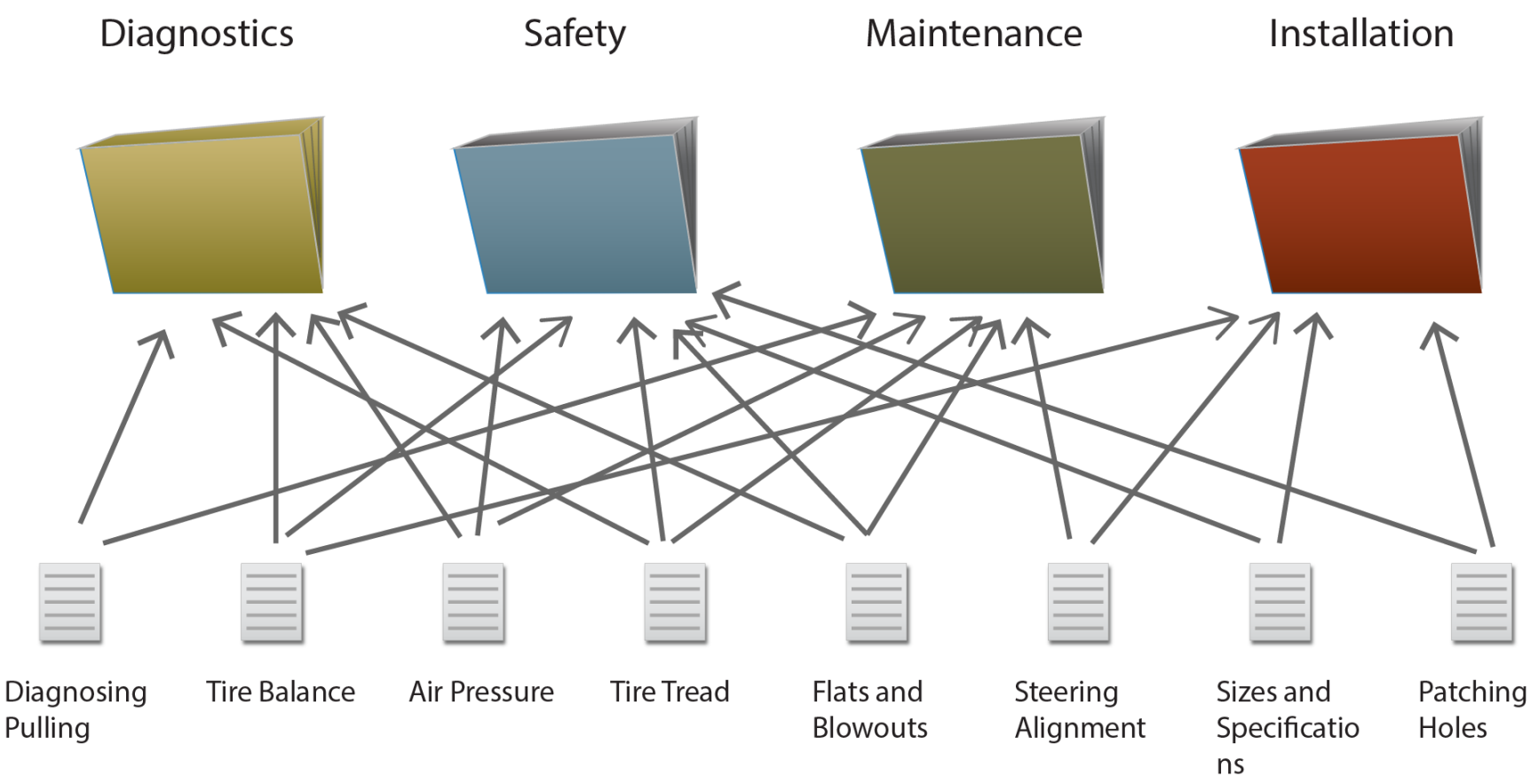
How do you help users find the right topic quickly and efficiently among hundreds of help topics? Is there a right way to organize content to optimize its findability?



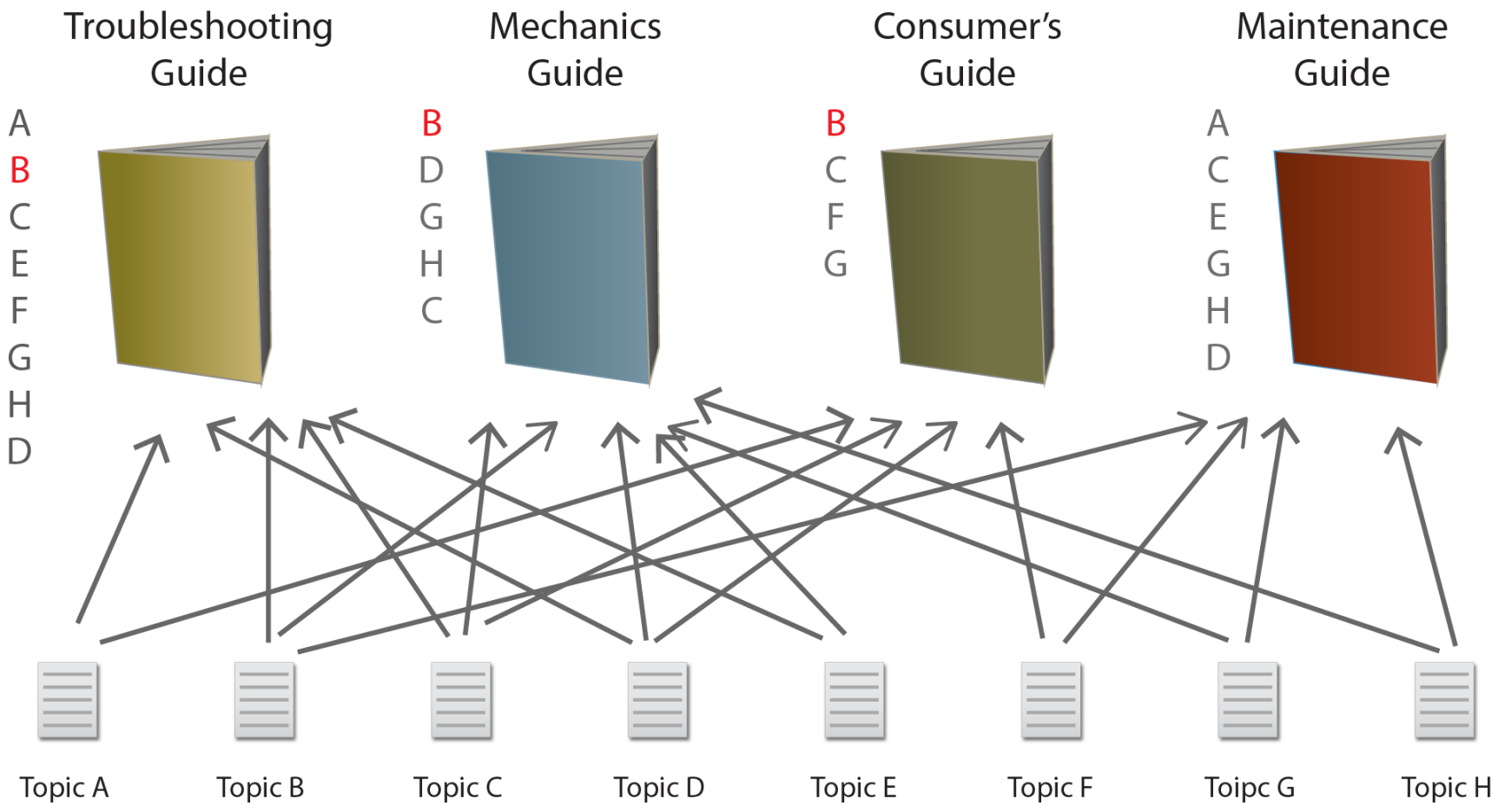
If topics were clearly distinct and easy to separate, we wouldn't have a challenge in organizing content.



But content doesn't fit neatly into separate and distinct folders. Topics overlap and defy categorization. They can be grouped in many different ways. It's hard for users to guess which way the author decided to group things. Content is like a platypus.



Some print-based solutions try to solve the organization problem by providing different outputs for different audiences and purposes. Topics can be re-used and repeated as needed in different groupings. For example, Topic B below gets reused in the first three books but not the fourth.



You can provide different web help files online to mimic the same print-based approach, but you still haven't solved the organization conundrum. You force the user to open one file after another, trying to guess which help file contains the right topic. It's the same book paradigm but online.

Troubleshooting Webhelp

A
B
C
E
F
G
H
D

Mechanics Webhelp

B
D
G
H
C

Consumer's Webhelp

B
C
F
G

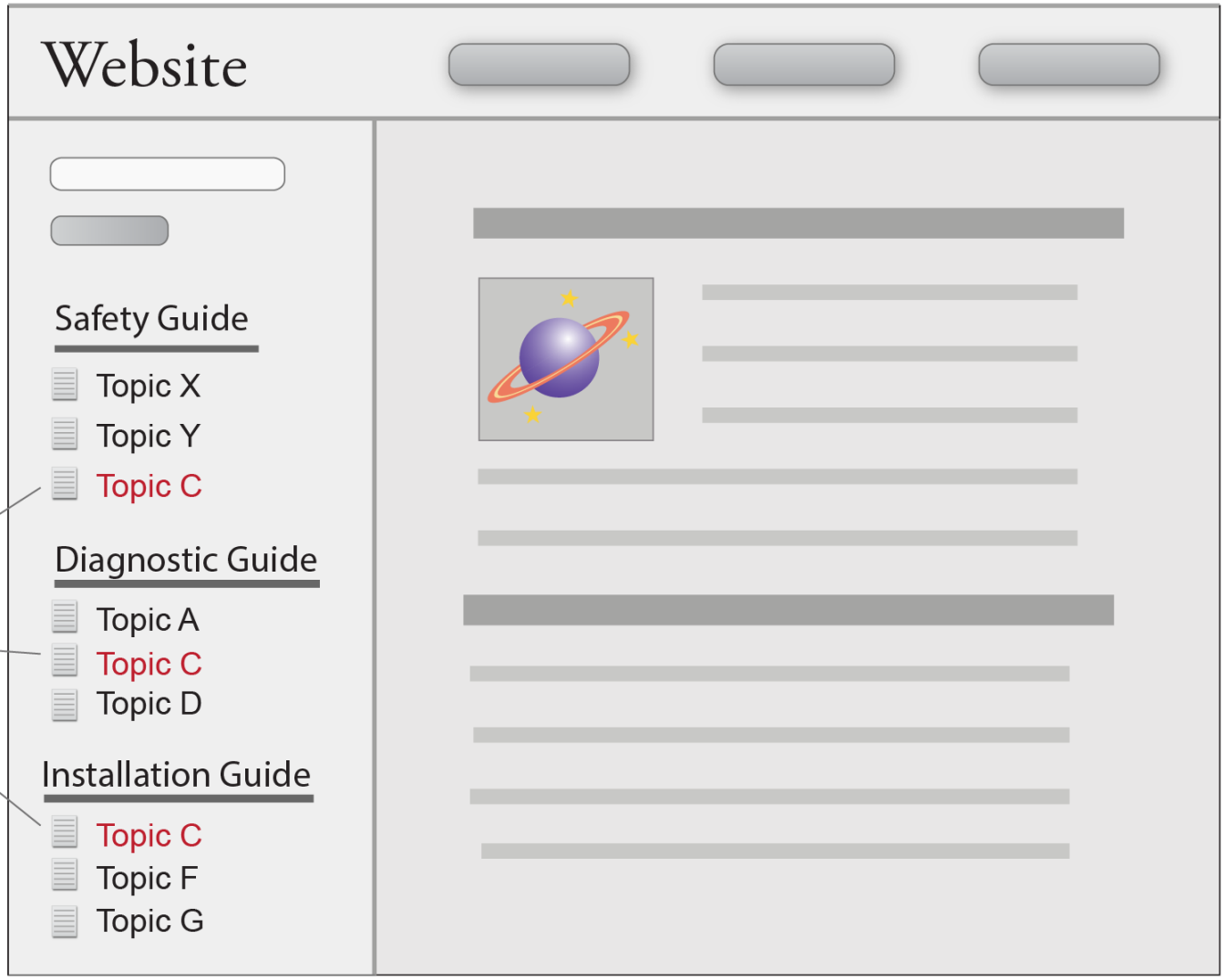
Maintenance Webhelp

A
C
E
G
H
D

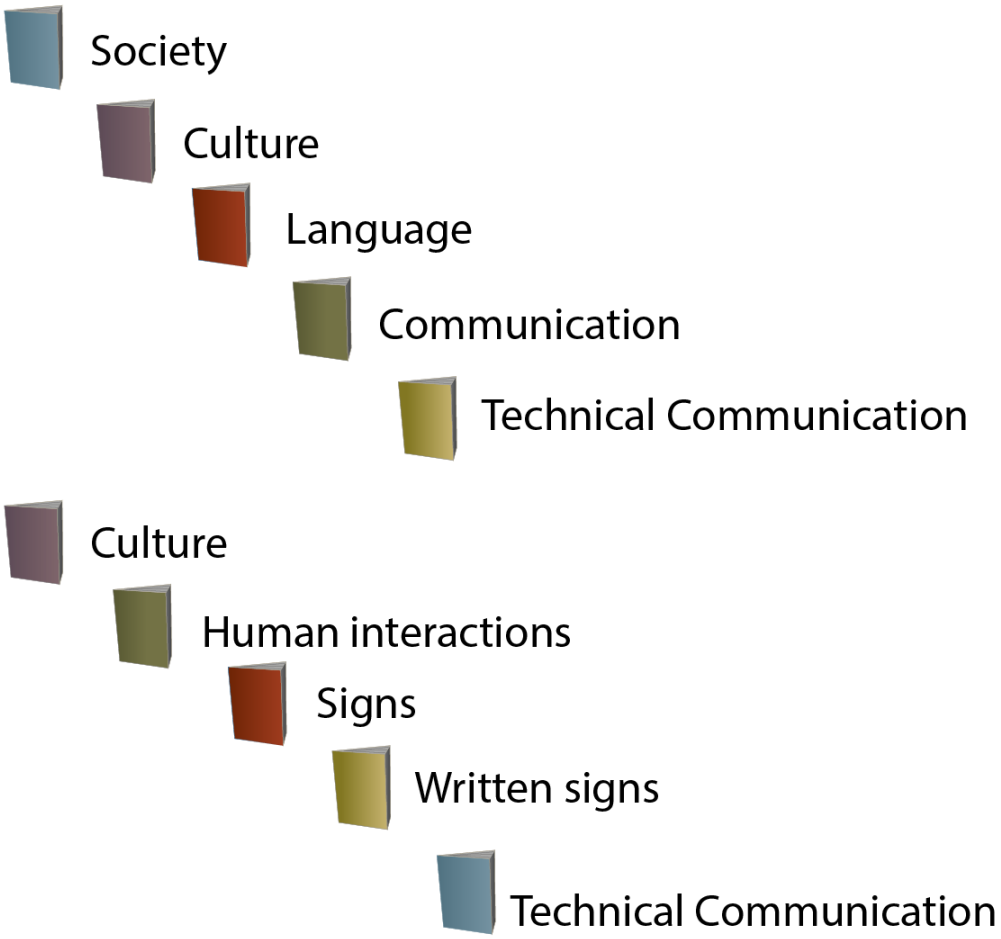
I'm not sure which guide would have the topic I want. Plus, I kinda want to search info from all of these files, not just one.

On a website (a web platform like Drupal or WordPress), you can't follow a print-like strategy of duplicating pages. Doing so ruins SEO, bloats content, confuses search results, and more. You could reference the same topic in different TOCs, but the TOC would be inconsistent for that topic.

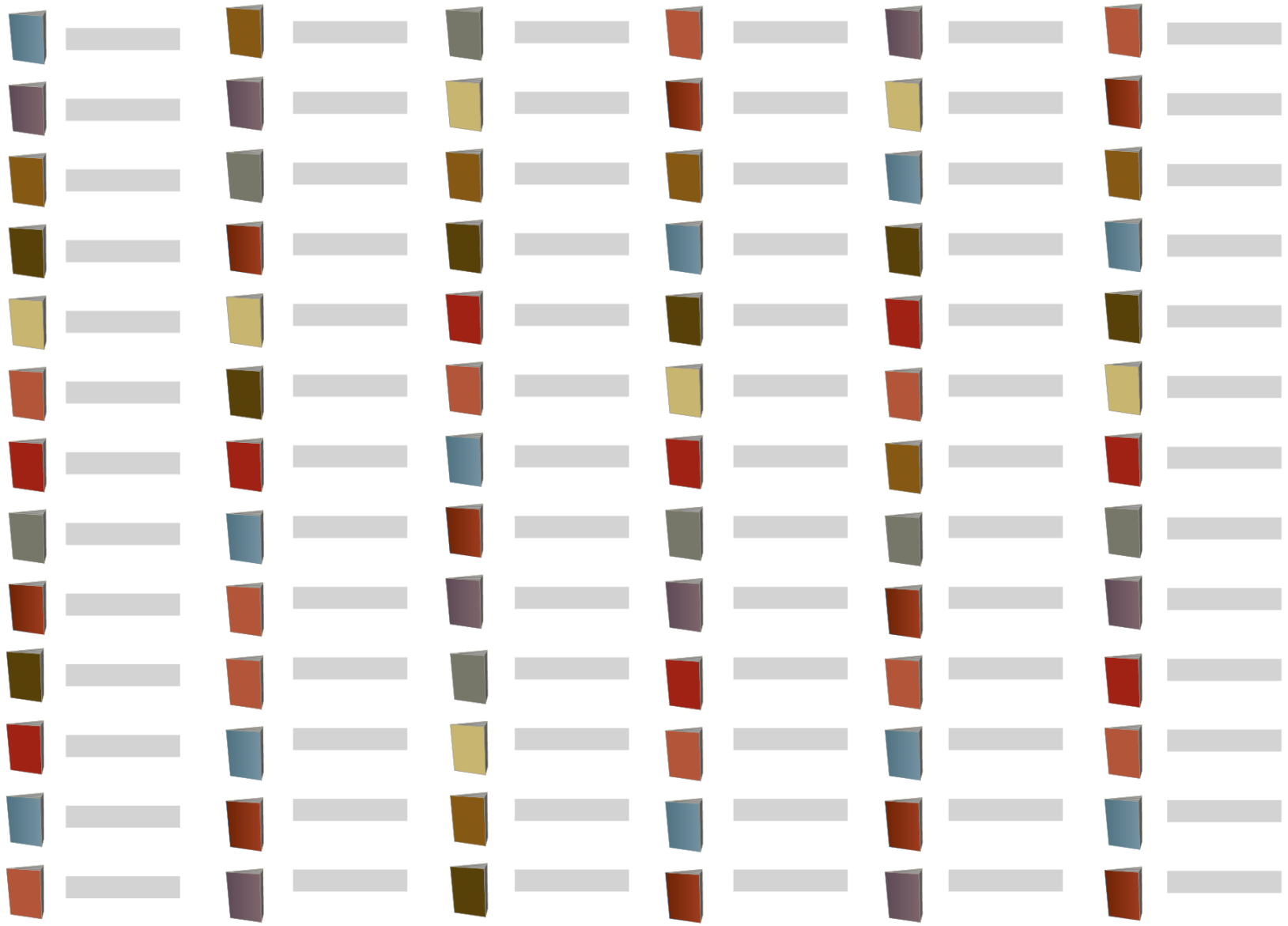
Which TOC appears when you view Topic C?



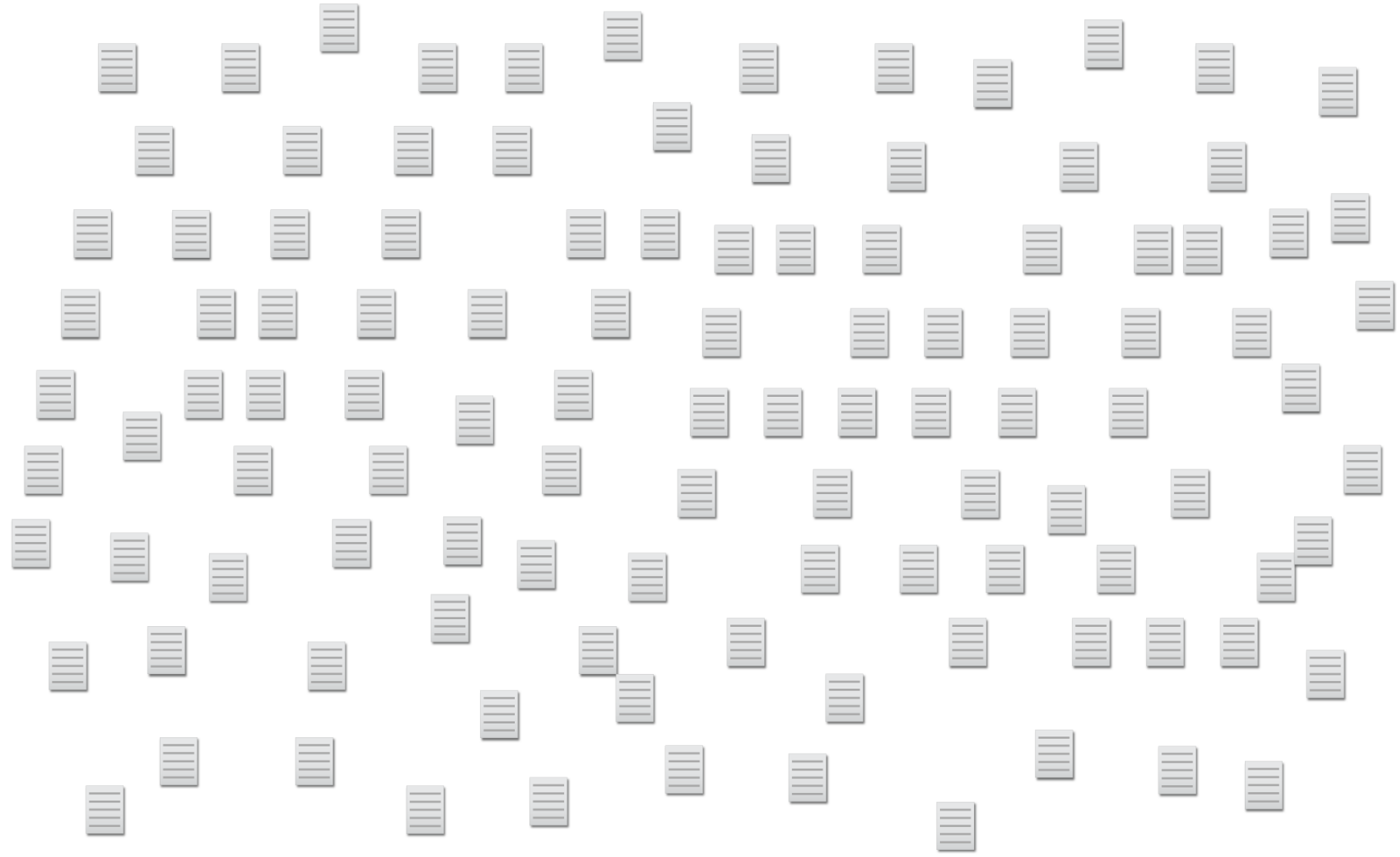
Sometimes writers resort to building a single massive table of contents (TOC) for all content. The problem with large-scale TOCs is that they force authors to nest content into so many hierarchical levels that the high-level groupings become abstract and meaningless to users.



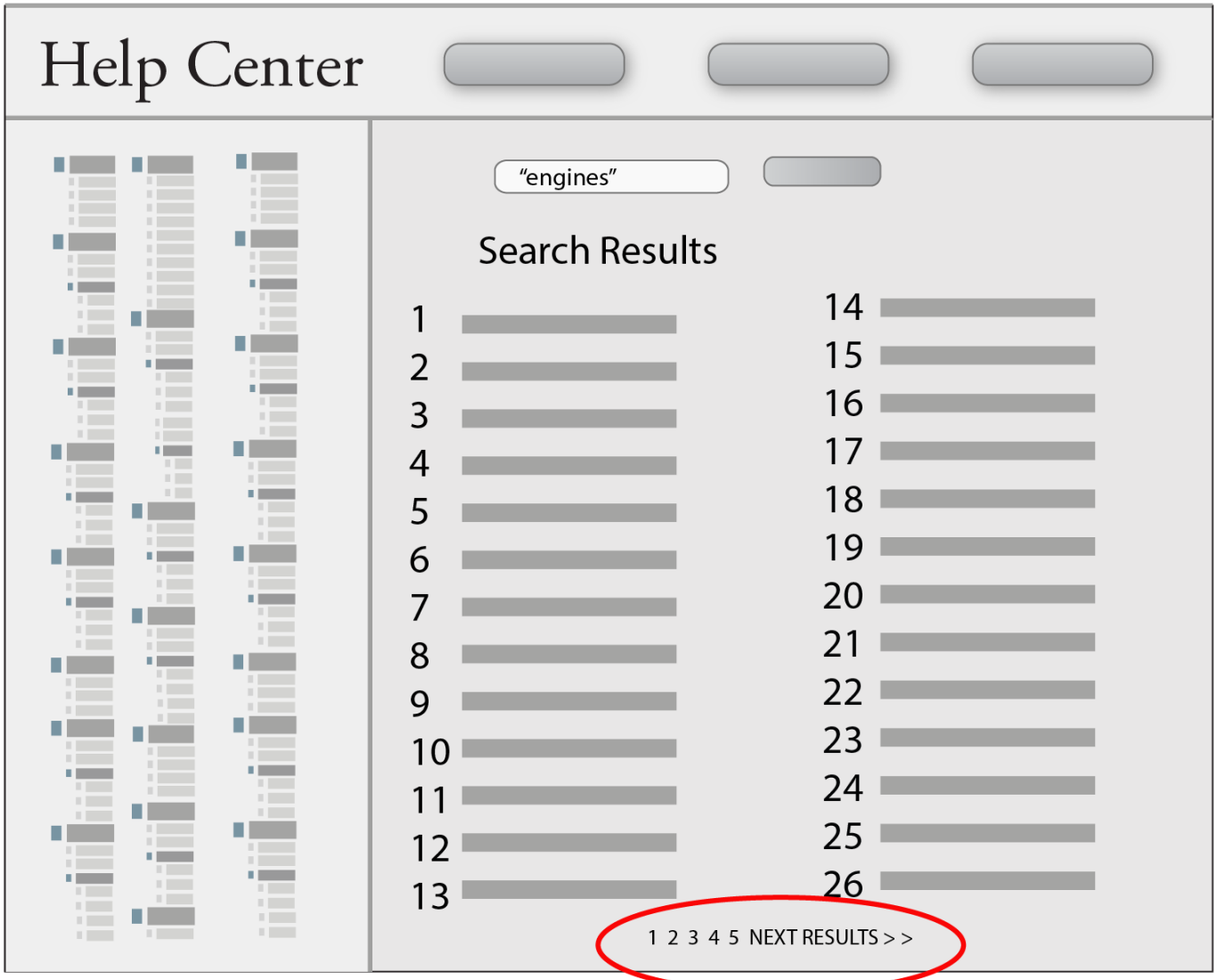
If you try to avoid deeply nested hierarchies in your TOC, you end up with a lot of shallow books to sort through, which makes it hard to see the TOC at a glance and get meaning from it.



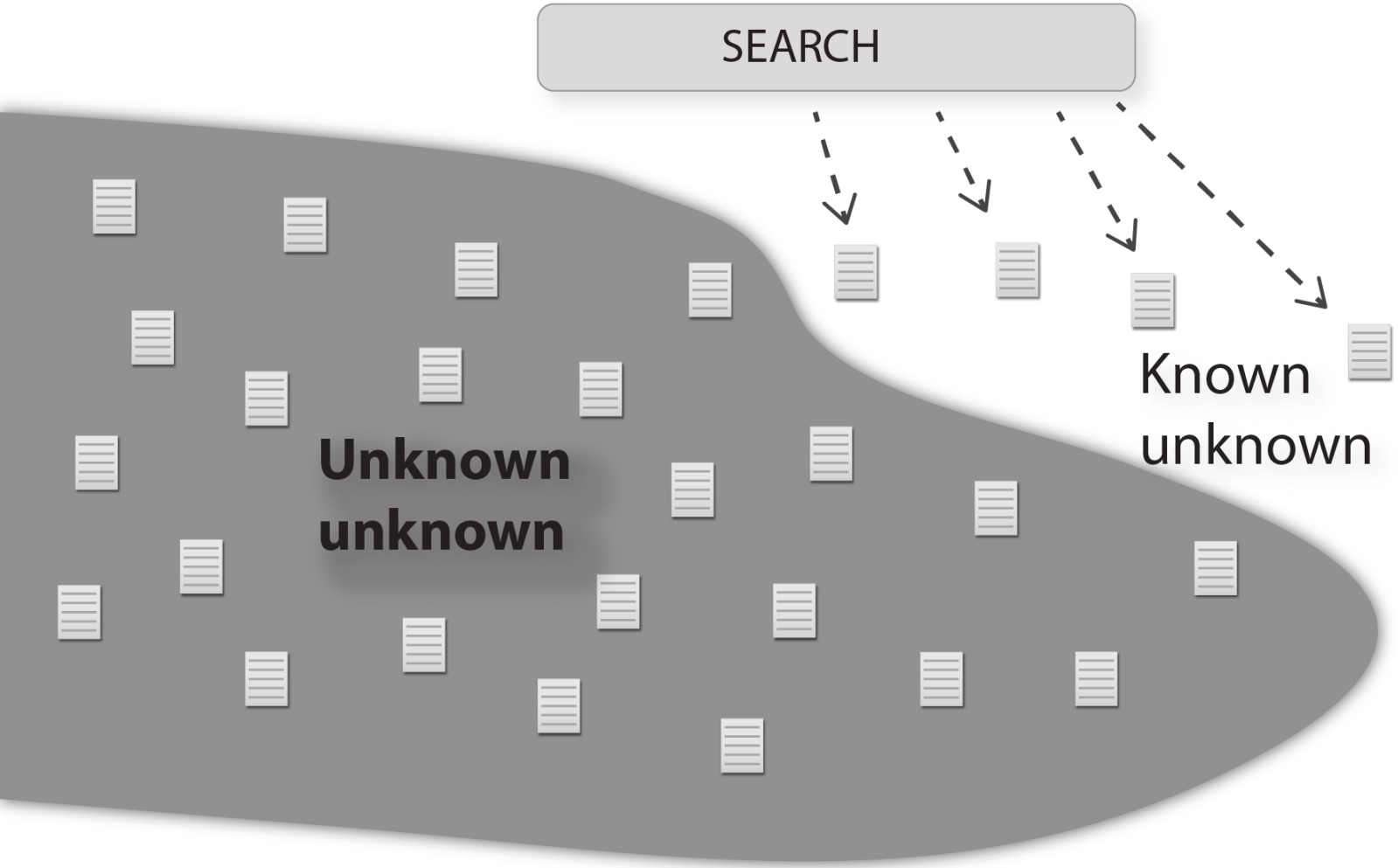
If you avoid TOCs altogether and forego any order at all, your content ends up flat. Users have to search to initiate a starting point. You have to build your navigation through links within each topic -- which is a good idea but tough and sometimes frustrating for users new to a system.



If you abdicate all findability to search, the search results are often a mess, returning hundreds of results or none at all. You have to guess what keyword strings will return the best matches. Searching becomes a trial-and-error experience repeated over & over with little variation.



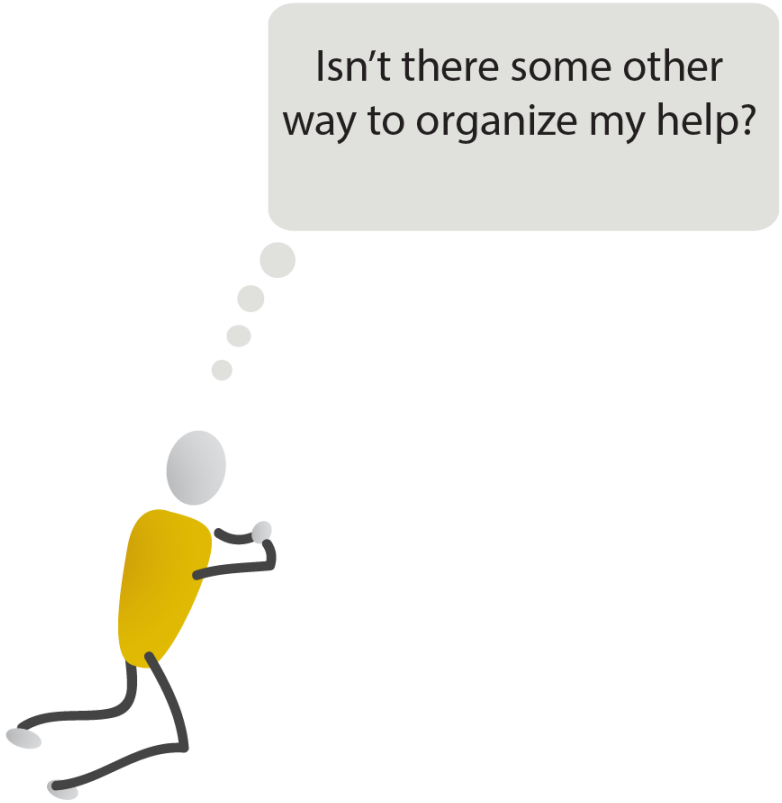
Search also doesn't help people become aware of gaps in their own awarenes. If users search only for terms and concepts in their vocabulary, it's harder discover new ideas. They don't find out what they don't know they don't know (the "unknown unknown").



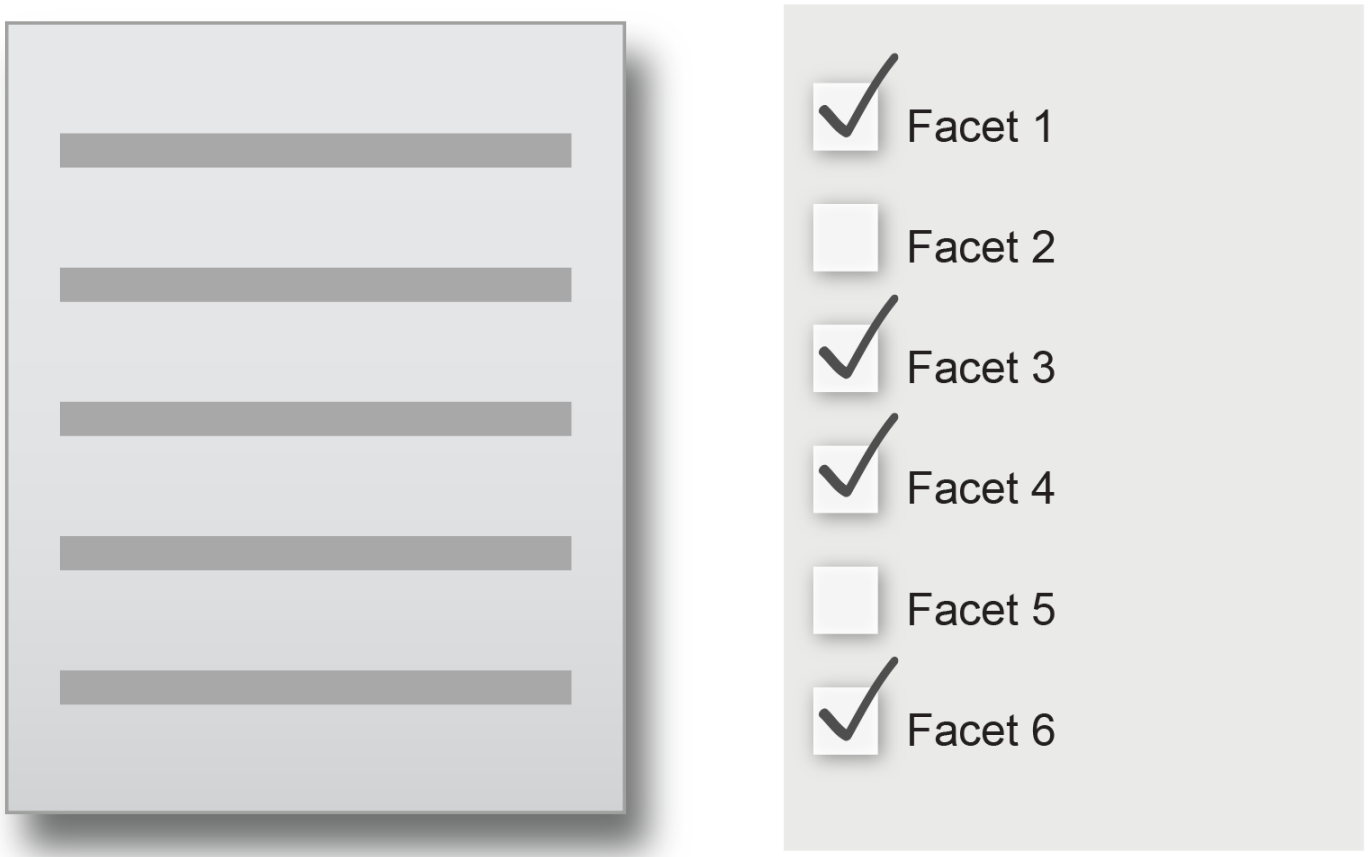
Ultimately, the TOC and traditional search pose problems for findability. We need a new model for help when writing for the web -- one that helps users find content quickly and easily.

Problem Summary

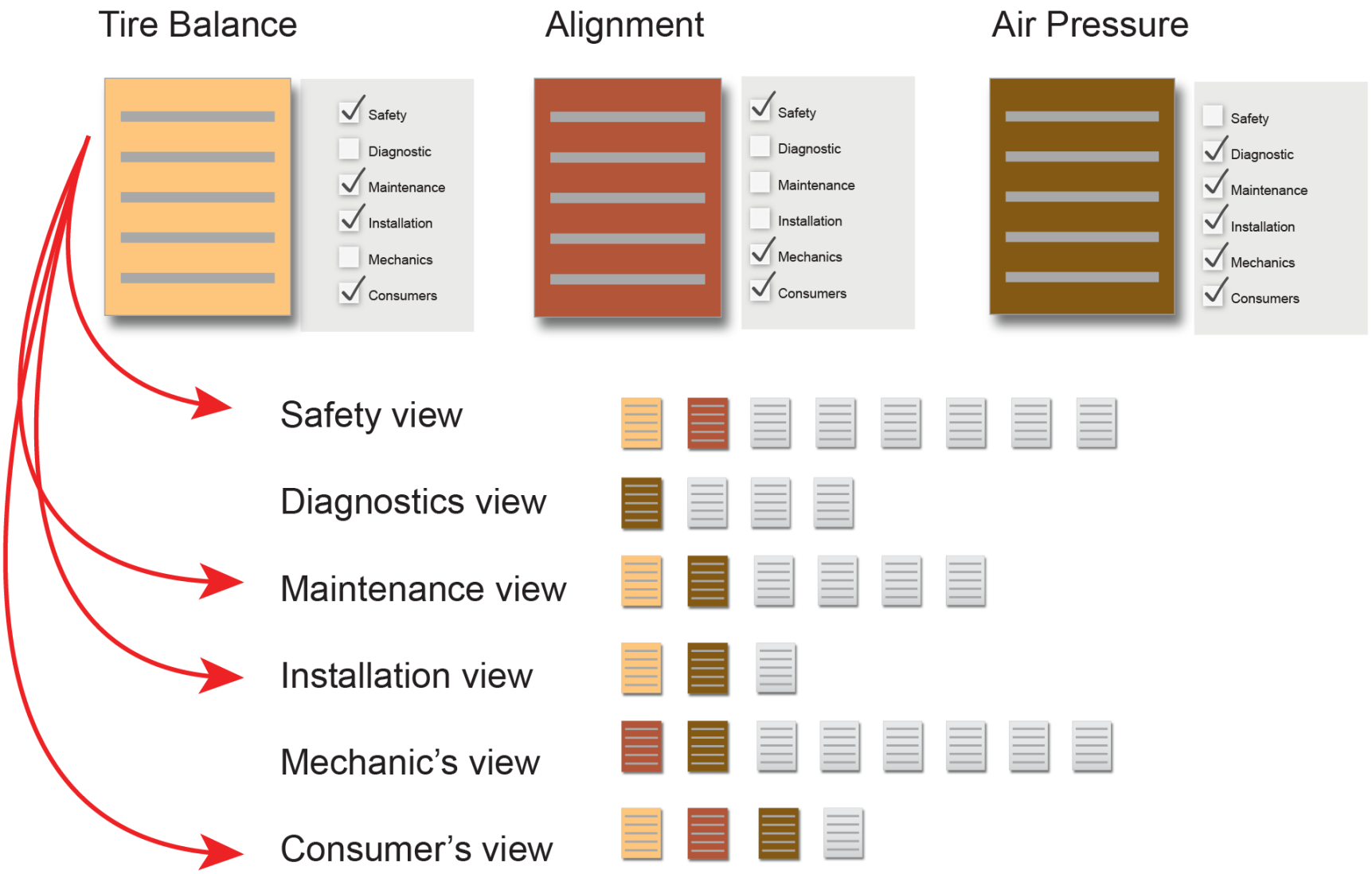
- Content can be logically categorized in many different ways.
- Paper-based paradigms for organization don't work on websites.
- Large-scale deeply nested hierarchies create categories with little meaning.
- Flattened organization provides too many options for meaningful understanding.
- Search doesn't provide a starting point for users new to a system.
- Search doesn't expose users to their un-known unknown.
- Search forces users to sort through massive amounts of results.



The epiphany: Drop the problematic TOC that forces a single path throgh the content. Instead tag your content with metadata and let users manipulate a selection of facets according to their interests. With this approach, content can be dynamically grouped just as the user wants.



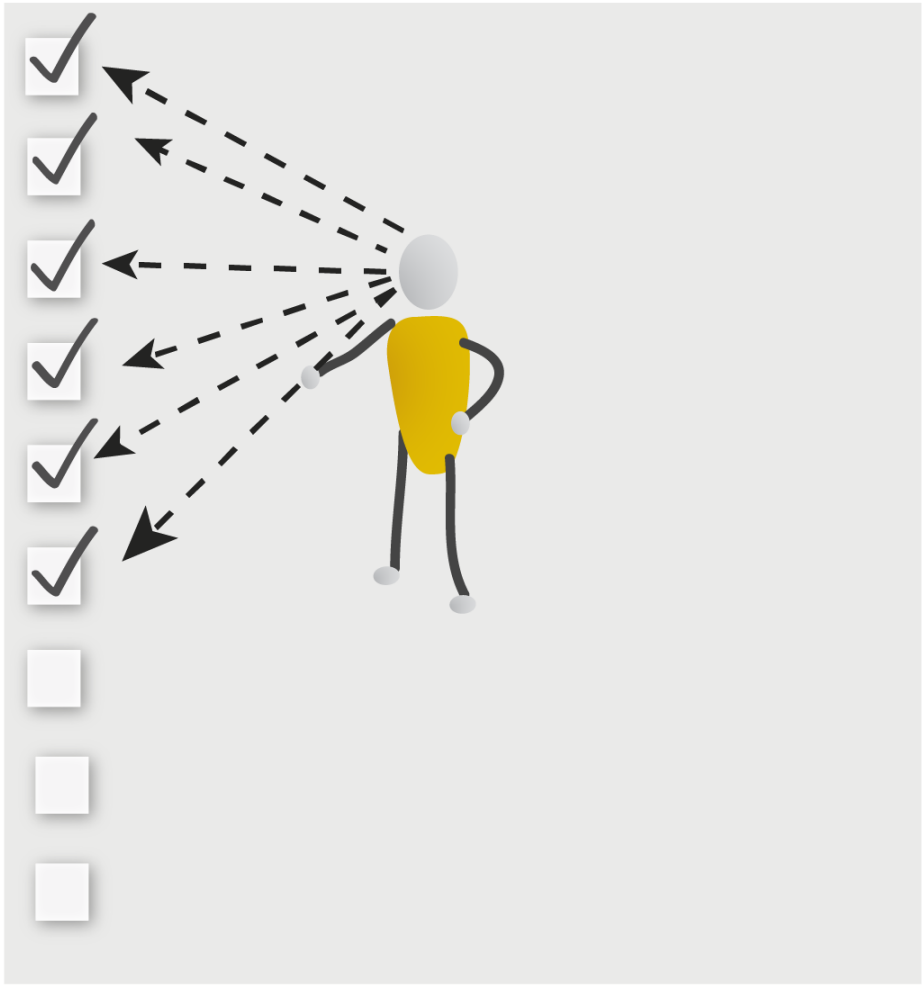
With this faceted appraoch, topics get grouped into different views based on the tag the author chooses to view. Peter Morville, a leader in information architecture, says faceted search is "arguably the most significant search innovation of the past decade."



The first step in creating faceted browsing is to create a taxonomy for your content. Then define the various views you want to pull the content into. Once you do these two steps, you can come up with a list of metadata for each topic so you can tag all your content with it.

- [x] Author
- [x] Date Published
- [ ] Recently Updated
- [x] Release Version
- [x] Related Screens
- [x] Role
- [ ] Popular Topics
- [x] Subject
- [x] Community Author
- [x] Release Notes
- [ ] Troubleshooting
- [x] FAQ
- [ ] Video
- [x] Forum post
- [ ] Blog article
- [x] PDF
- [x] Event type
- [ ] Case Study

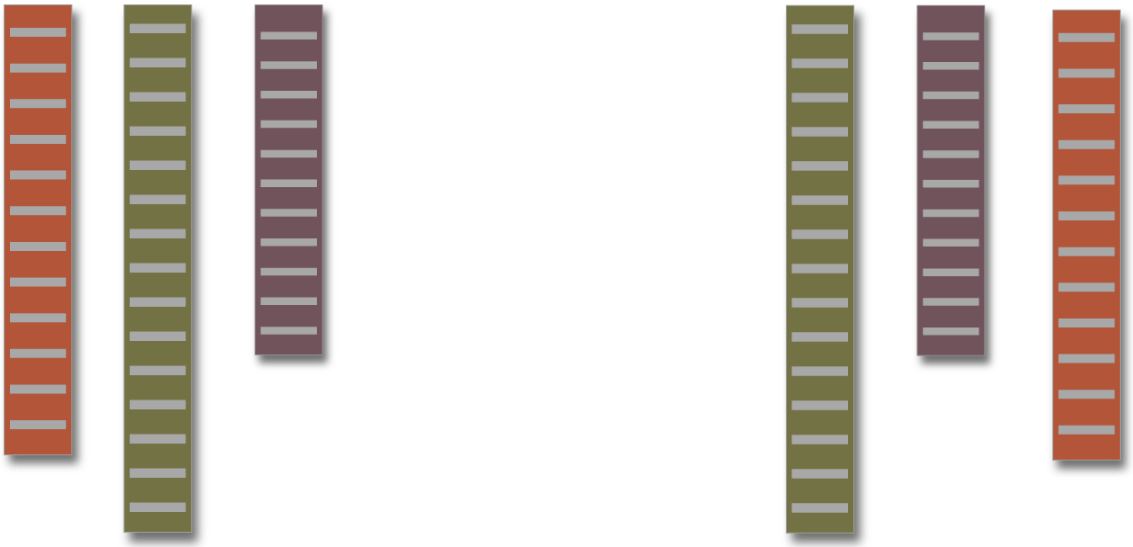
The facets need to come from your user's terminology and familiarity. It doesn't do much good to impose an institutional taxonomy or an unfamiliar language if your users aren't going to navigate by it. Try to pull ideas from search queries to get a sense of your users' language and interests.



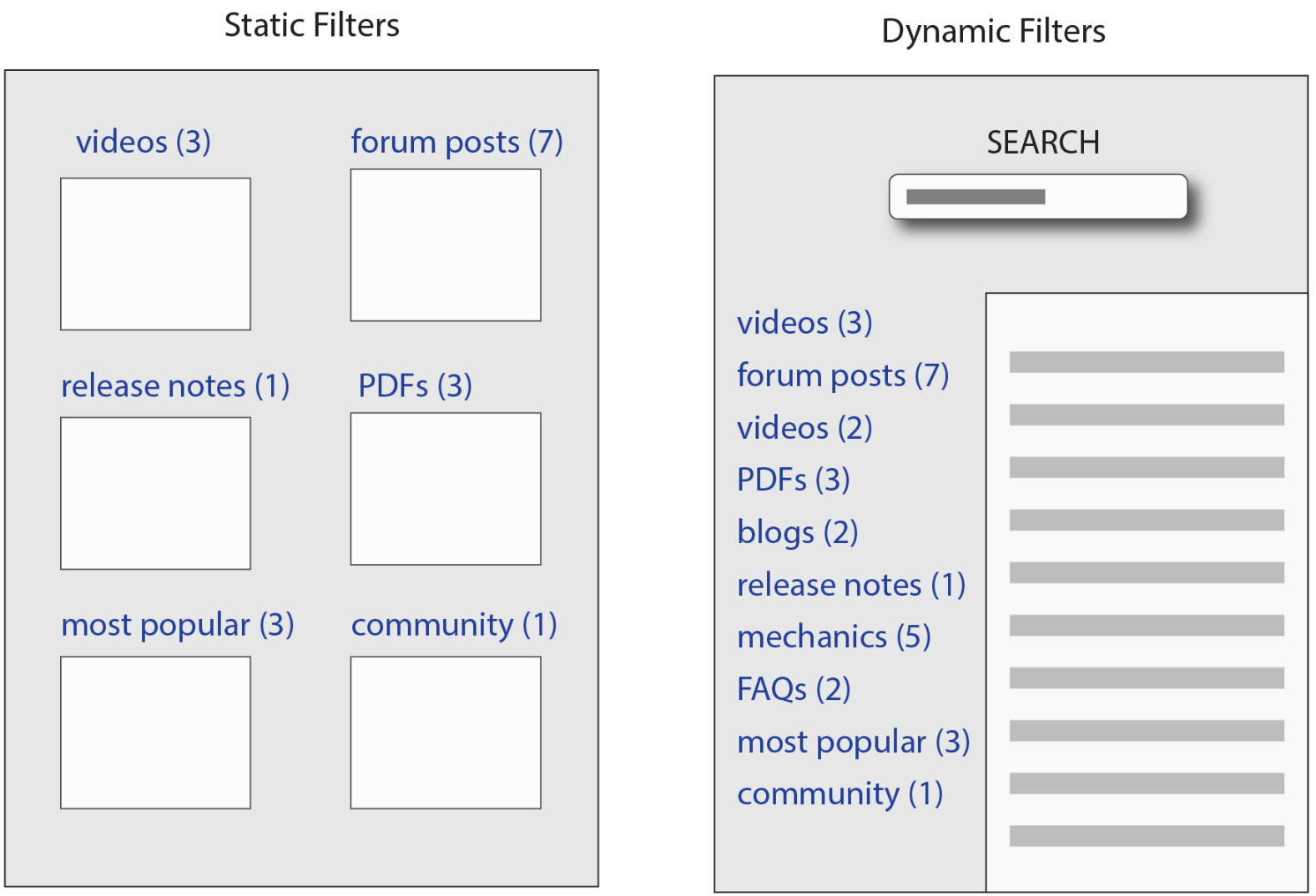
Content needs to be chunked small enough so all the various topics can be recombined in different ways. Large topics provide fewer opportunities to be pulled into different facets. Content still, however, needs to be meaningful enough to stand on its own.

Small chunks

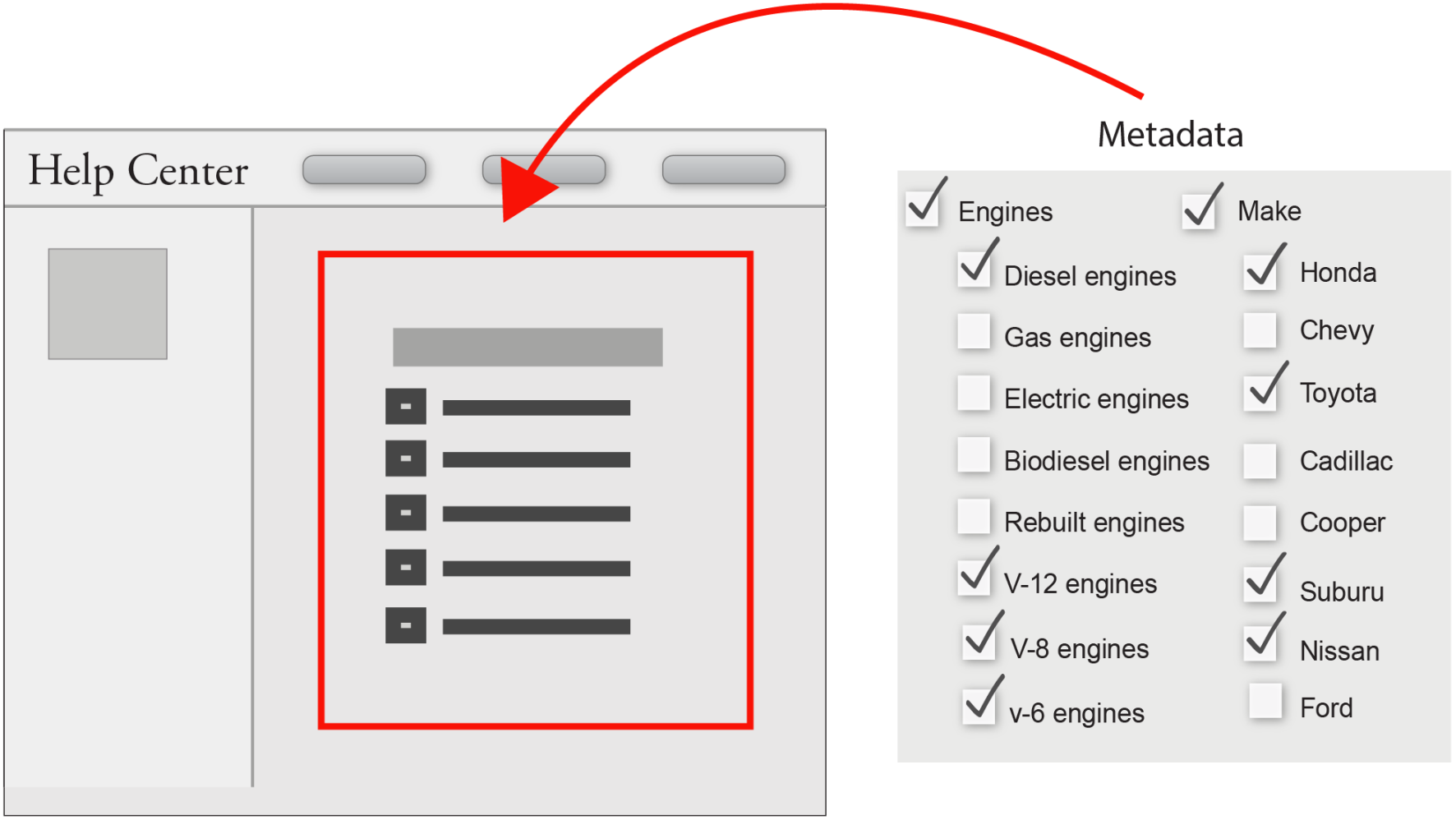
Large chunks



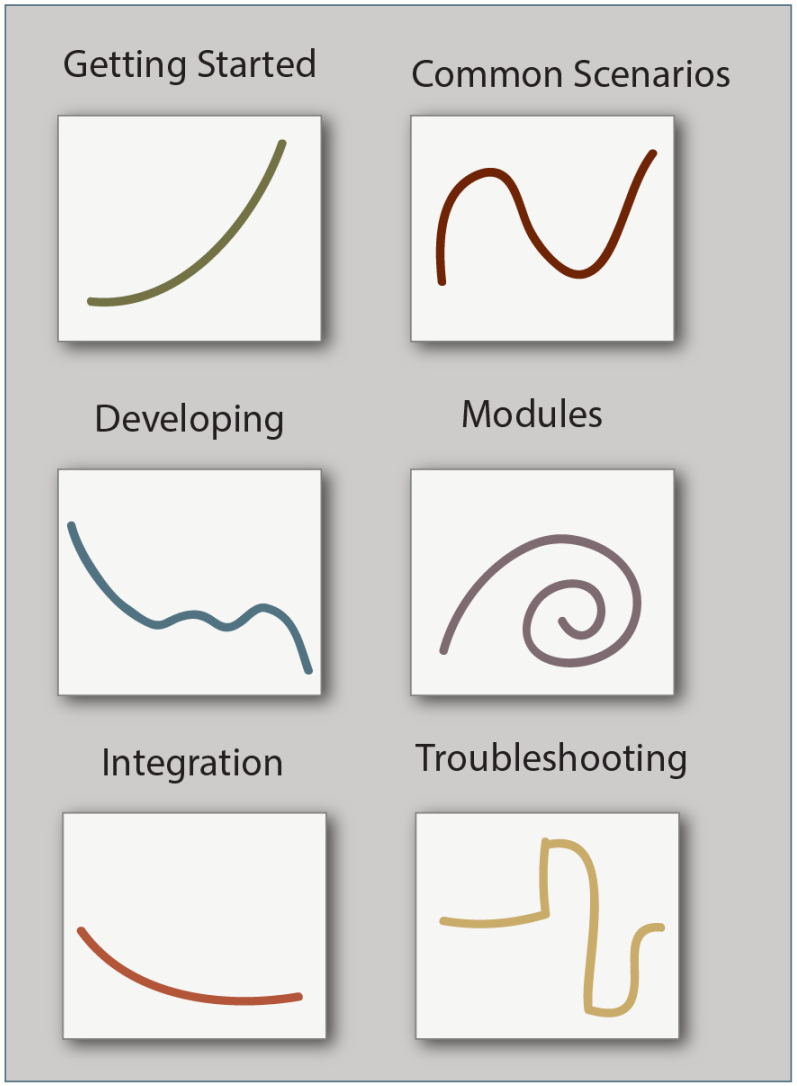
Your metadata can provide *dynamic* facets, which appear based on user searches for a keyword, or *static* facets, which present a fixed query to a system.

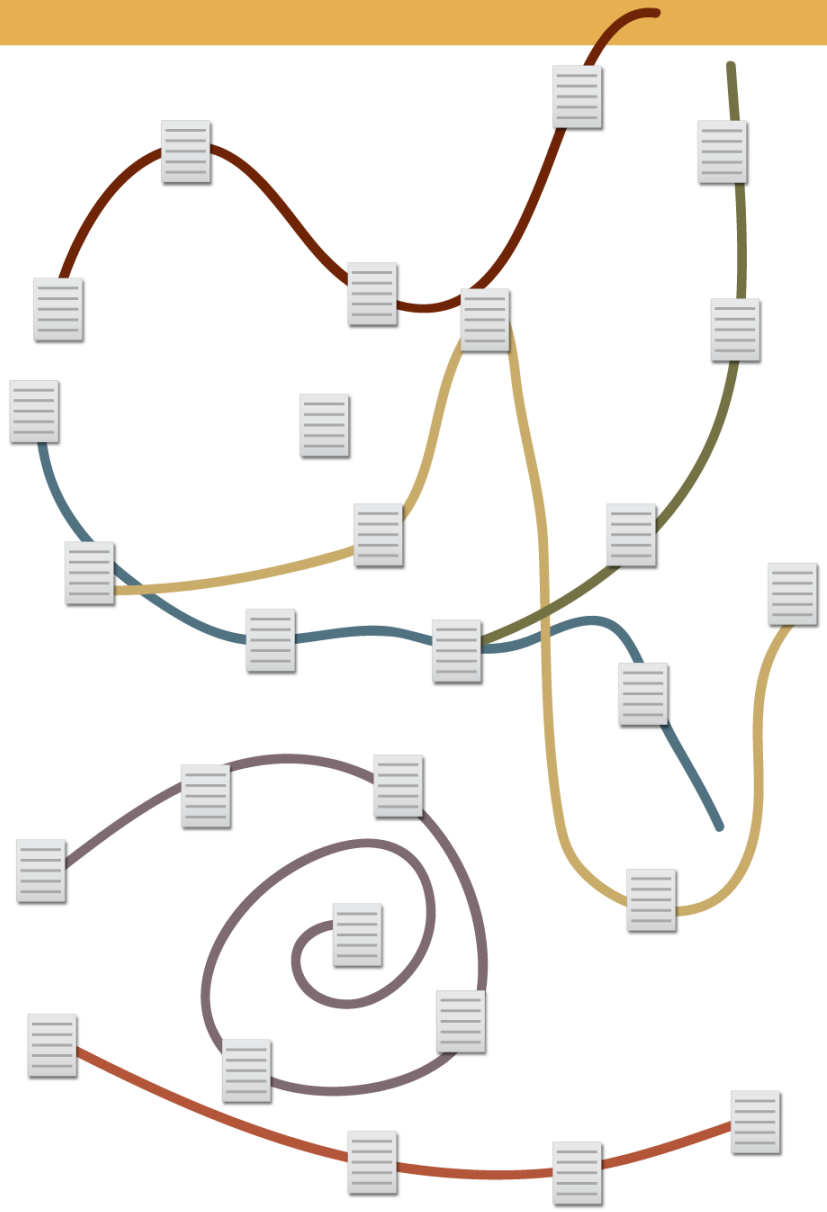


Through "metadata choreography," as Seth Earley calls it, you can define what content appears in each part of your site. This space below will show all diesel engine Japanese cars with V12, V8, and V6 engines because I filtered the metadata here this way.



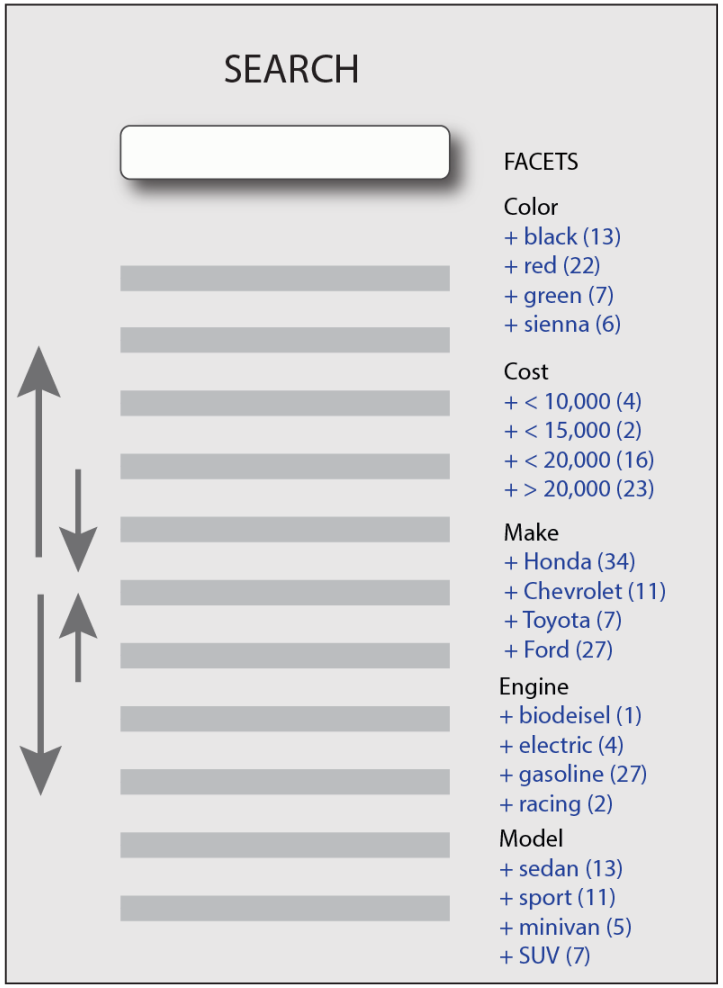
You can also create portal pages that roll up specific topics in different groupings. These landing pages serve as guides through the content, offering different navigation options through a list of topics without imposing a single fixed order.





The application interface can be another type of portal to your help content. Links in the interface can connect to all topics tagged with that interface screen name.

With search, dynamic facets allow you to narrow the results until you find what you want. You start broad and then get more narrow. If you narrow the results too far, you can easily clear the facet to expand the search results. You can essentially zoom in and out with one search query.



Color
+ black (13)
+ red (22)
+ green (7)
+ sienna (6)

Cost
+ < 10,000 (4)
+ < 15,000 (2)
+ < 20,000 (16)
+ > 20,000 (23)

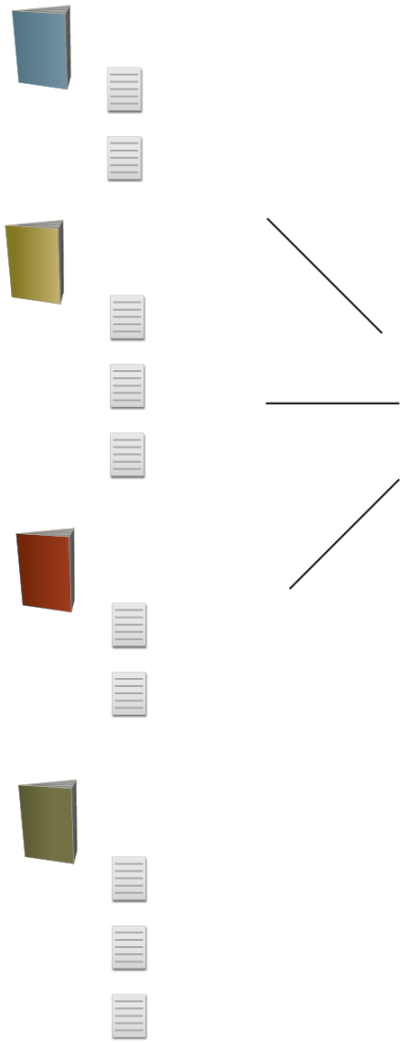
Make
+ Honda (34)
+ Chevrolet (11)
+ Toyota (7)
+ Ford (27)

Doc Type
+ video (3)
+ forum (12)
+ blog (4)
+ documentation (32)

Engine
+ biodiesel (7)
+ electric (5)
+ gasoline (3)
+ racing (15)

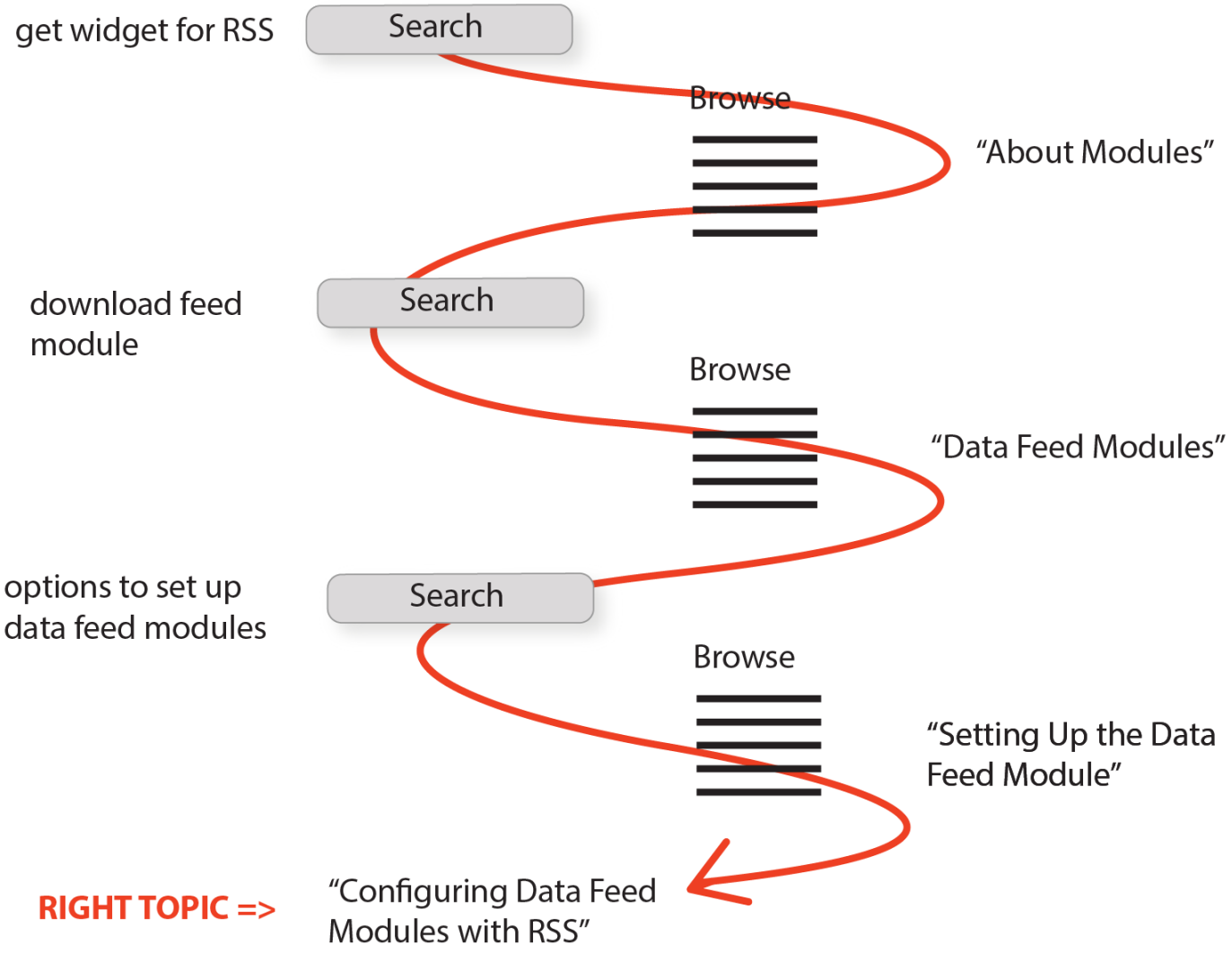
Model
+ sedan (3)
+ sport (12)
+ minivan (4)
+ SUV (32)

Faceted search helps narrow down the massive TOC. You only see facets relevant to your search, so you get a zoomed-in and partial view of the TOC as it relates to your search query. In this way, you get the best of both worlds: a relevant and small-scale TOC *and* a single site for all your content.

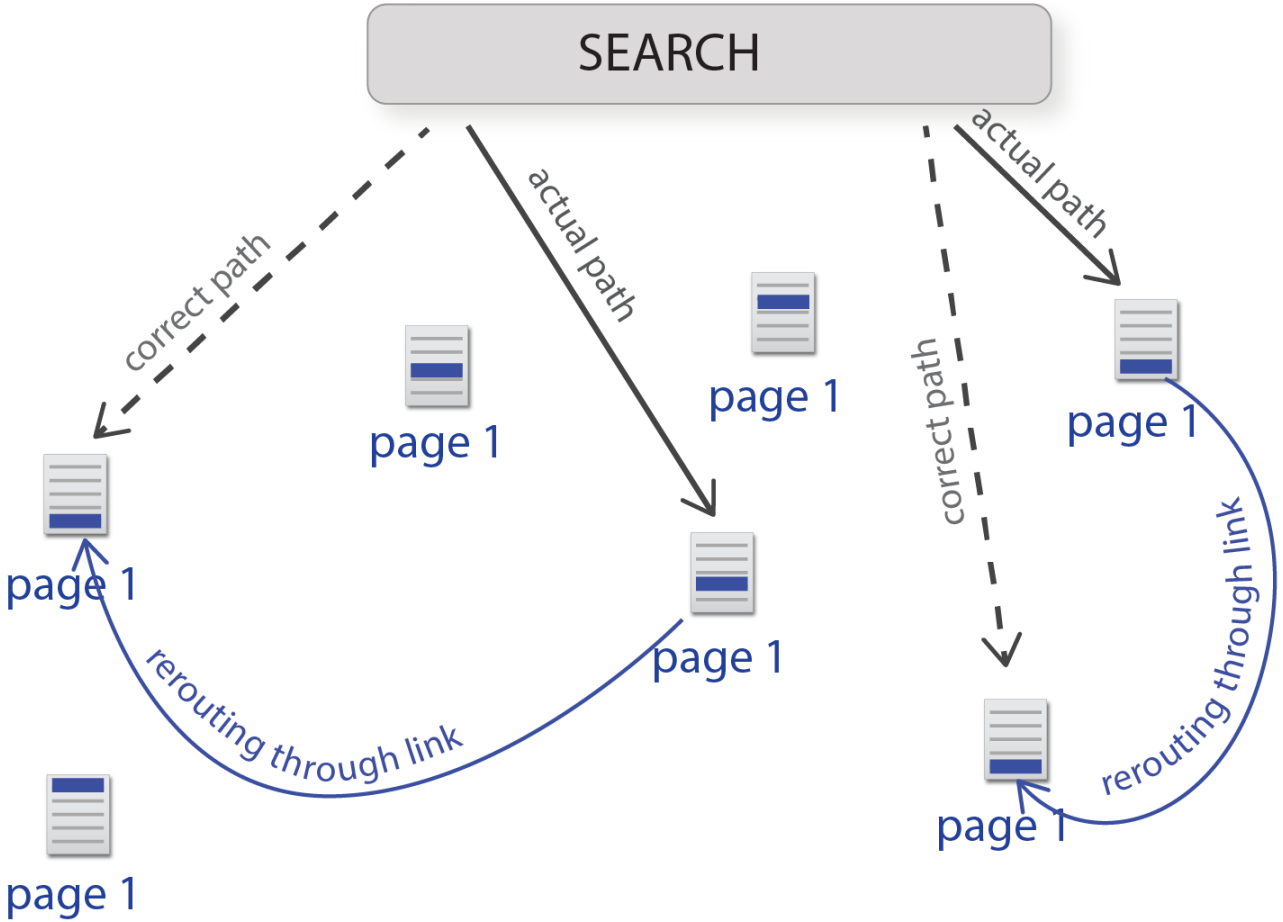


Only this part of the navigation appears because only these pages have keyword relevance with your search.

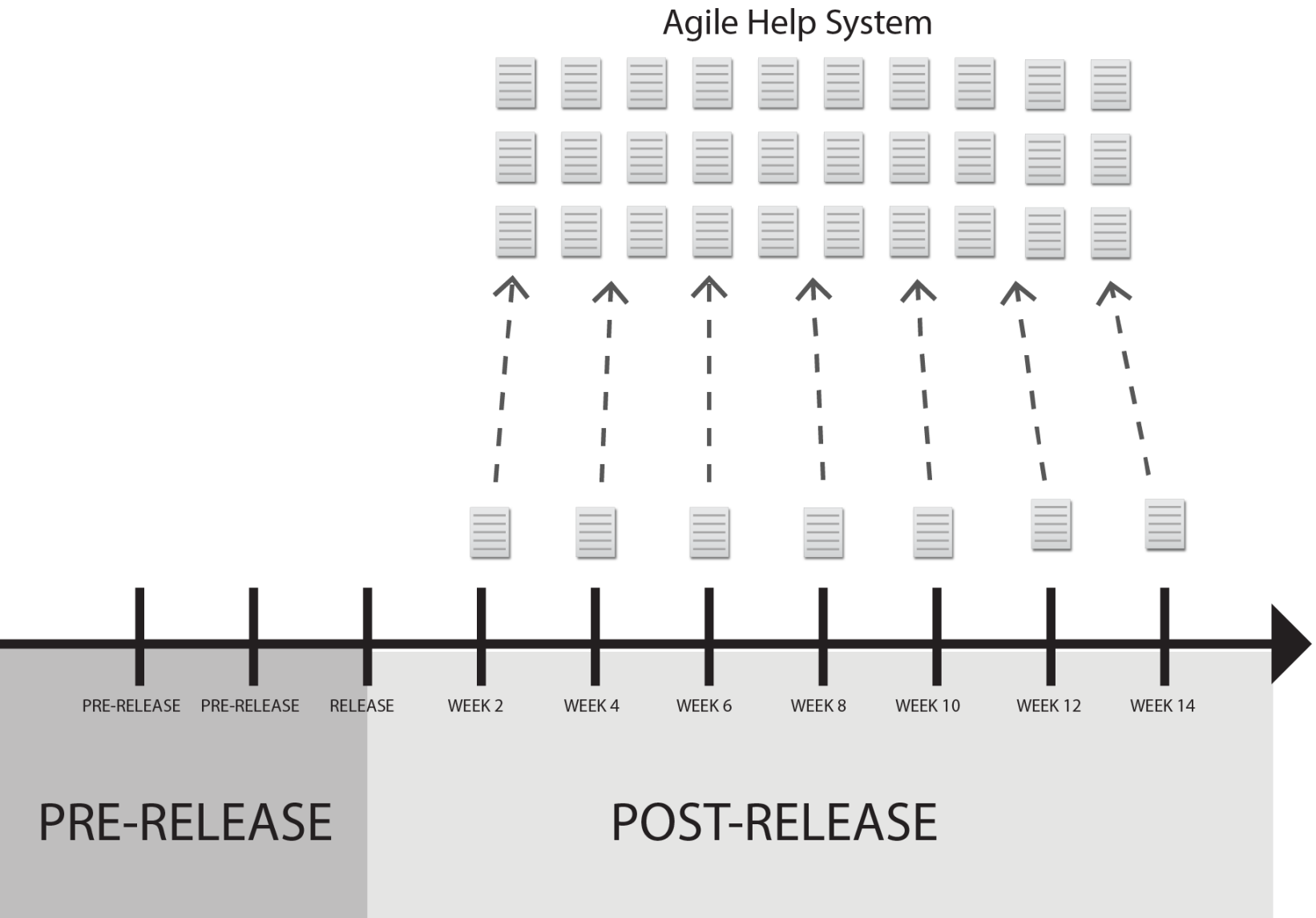
Searching and actually browsing work great together. The two activities inform each other. Search lands you in one place, you browse and learn more vocab and concepts, and then you search more intelligently. This lands you in a more precise area to browse, and so on.



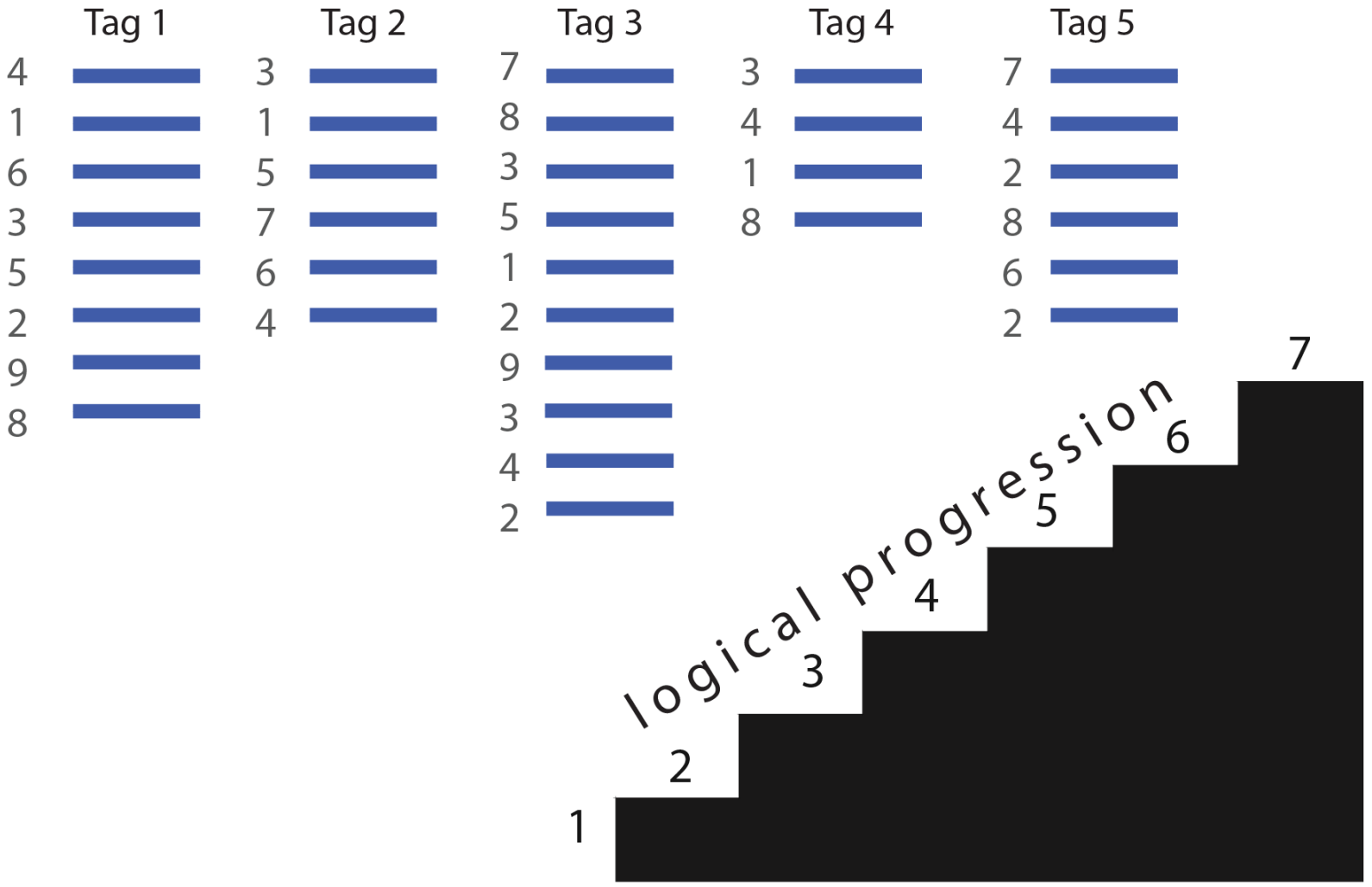
To further help users find the right topics, provide links within topics (both as related links and inline links). Links within a topic can help reroute users to the correct topic (what Mark Baker calls the "last mile of findability"). These in-topic links can be auto-generated based on your metadata.



When you're constantly adding to your help (as is the case with agile publishing environments), you need an easy way to keep adding to the system. Through this metadata approach, your topics become "plug and play" with the help system.

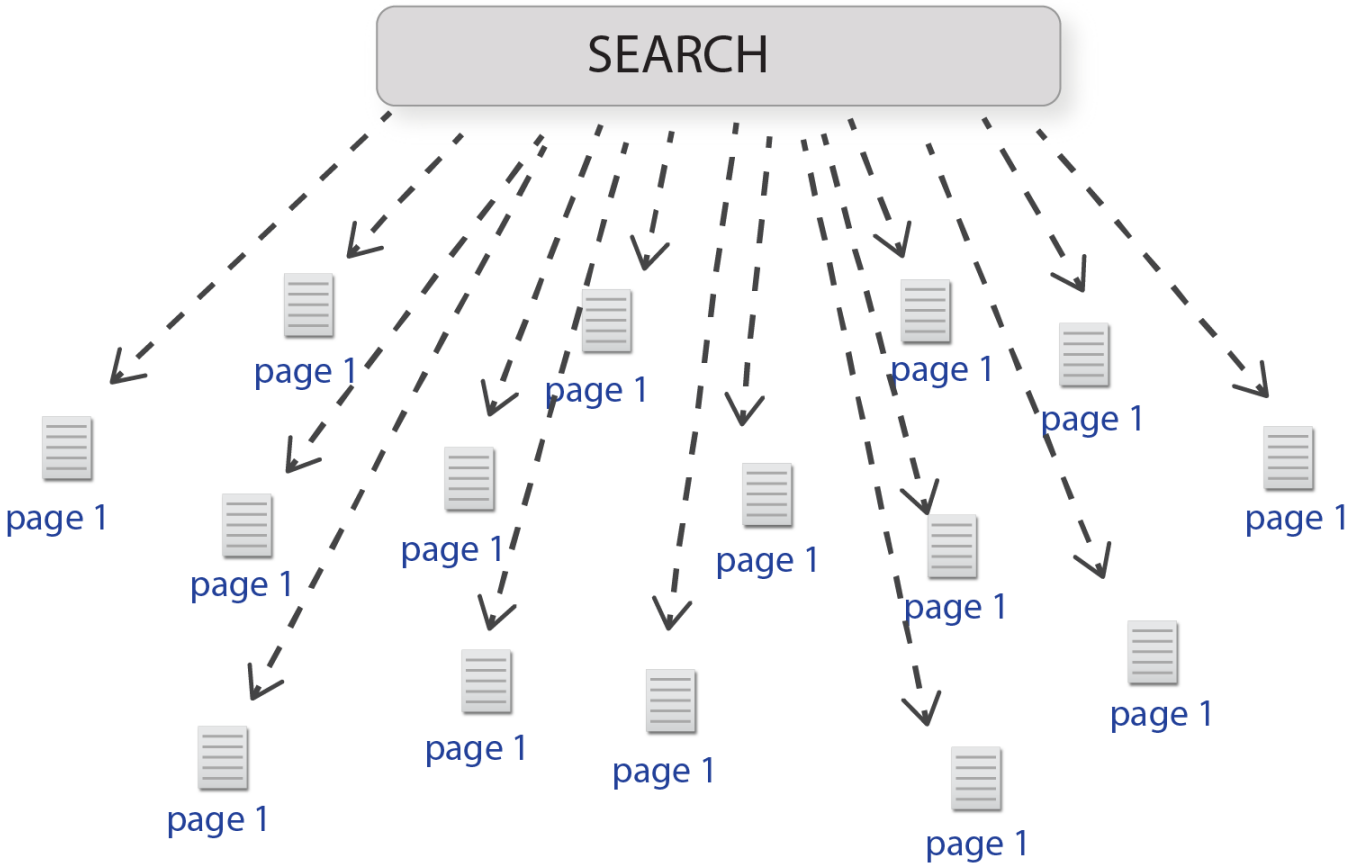


One problem with tagging is that you end up with a lot of lists of topics without much hierarchy or order to them. This non-linearity works great when users are searching for specific questions but not so well when users are trying to learn a new system.



You don't have to ditch the TOC entirely. James Kalbach says, "Navigation provides a narrative for people to follow on the web." But use that linear navigation only when your content has a sequential, story-like nature to it.

As you deemphasize the linear sequence of a TOC, you put search more in the foreground. In order for search to succeed, every page needs to be a starting point in itself (a "page one," as Mark Baker says). When each topic is autonomous in a meaningful way, topic order doesn't much matter.

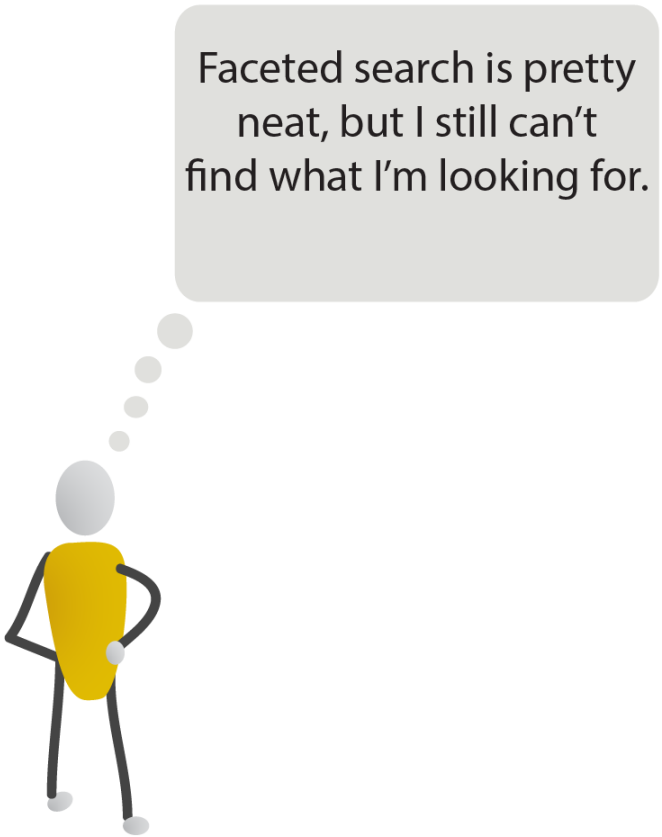


Metadata and facets go a long way toward overcoming some of the challenges to content organization.

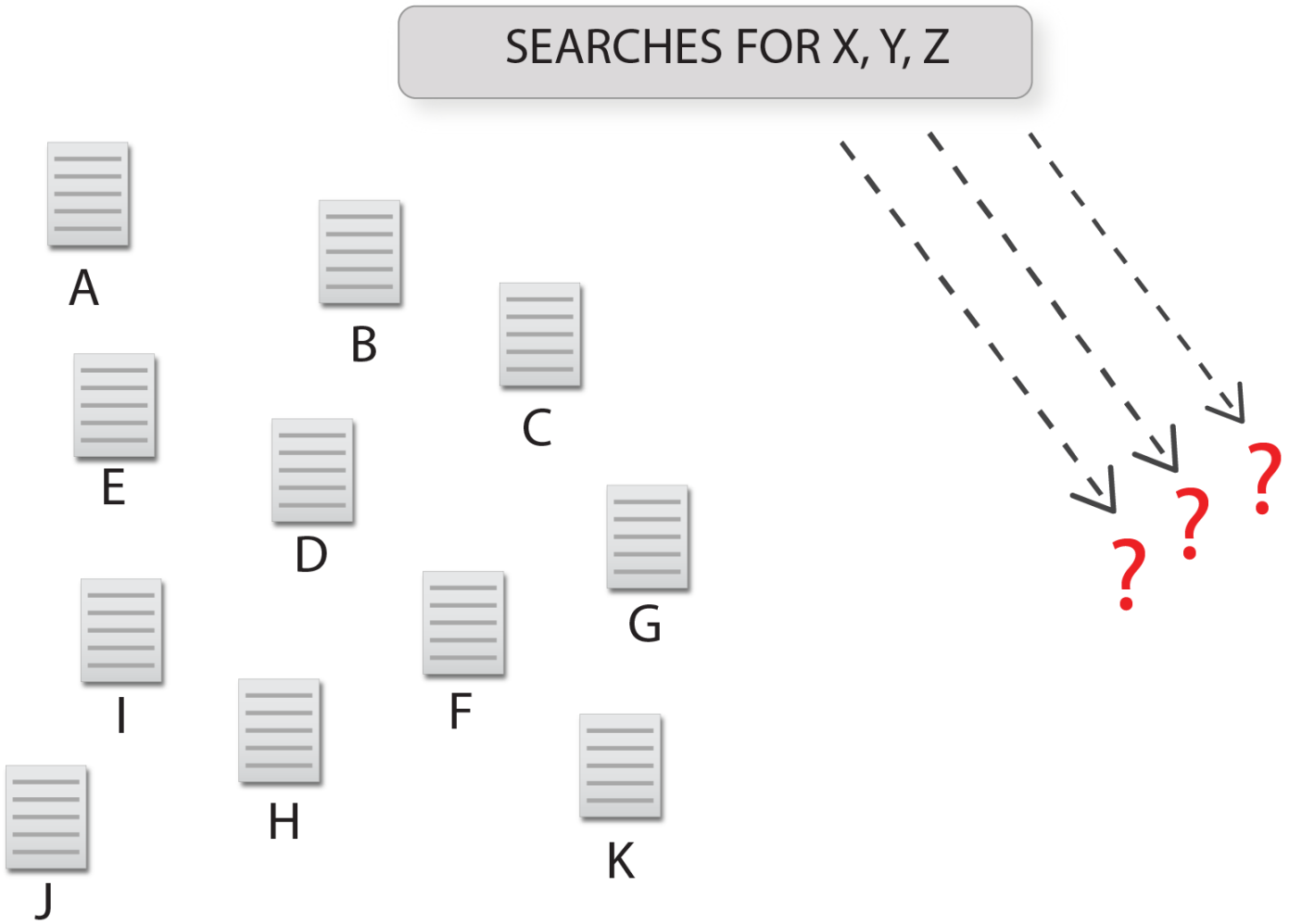
Solution Summary

- You can create multiple views to categorize the same content in many different ways.
- A single topic or component can be repurposed into different views without duplication.
- Large-scale hierarchies are hidden from users.
- To overcome the flattened organization, users zoom in to a relevant scope from their search.
- Users can combine browsing and searching in intelligent ways. The browsing helps them discover new terms and concepts.
- Users can construct smart queries on the fly, adjusting the amount of results.

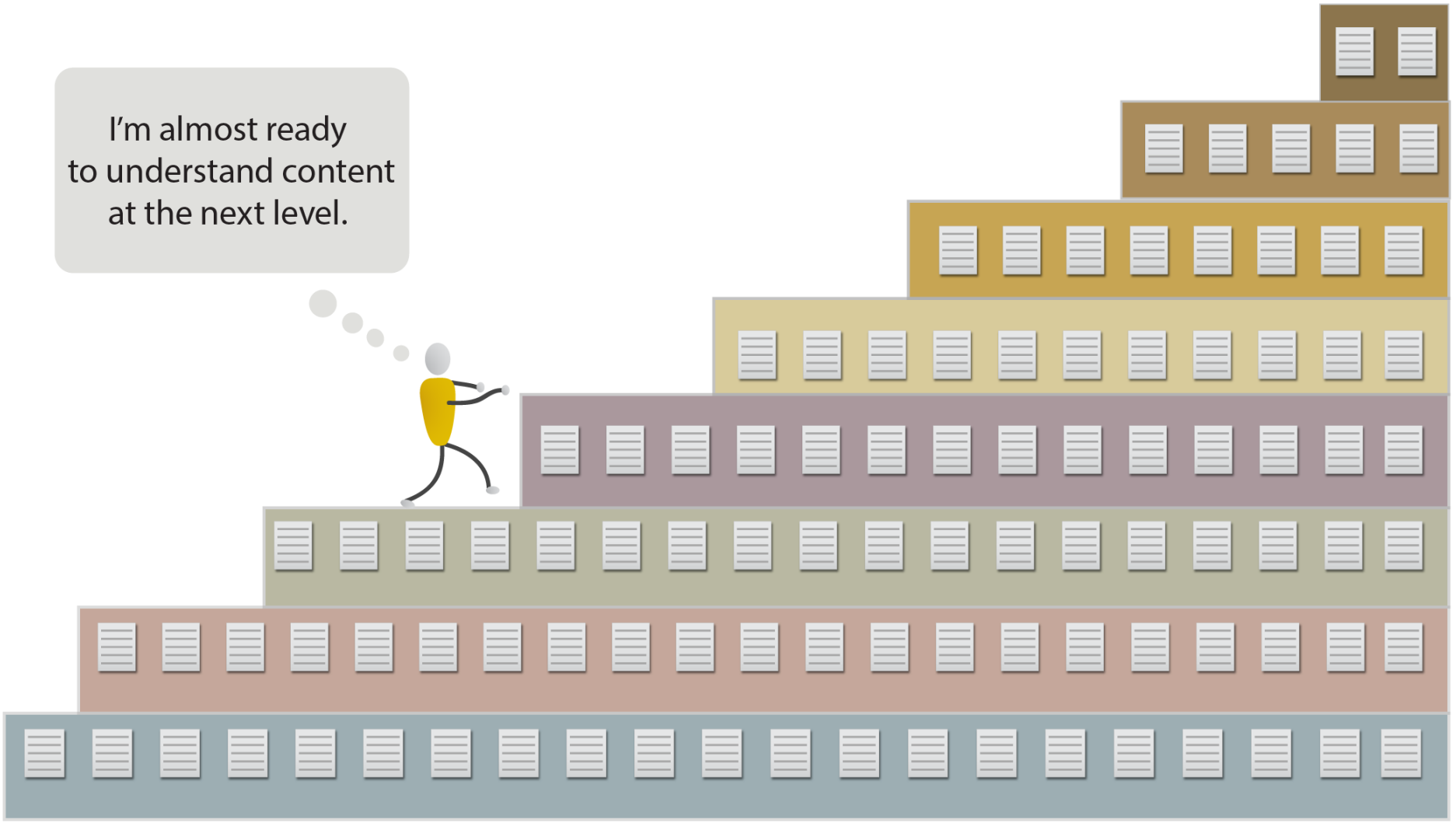
Even though faceted search and metadata are pretty powerful, they doesn't solve all the problems with content findability. There are other reasons why users can't find content.



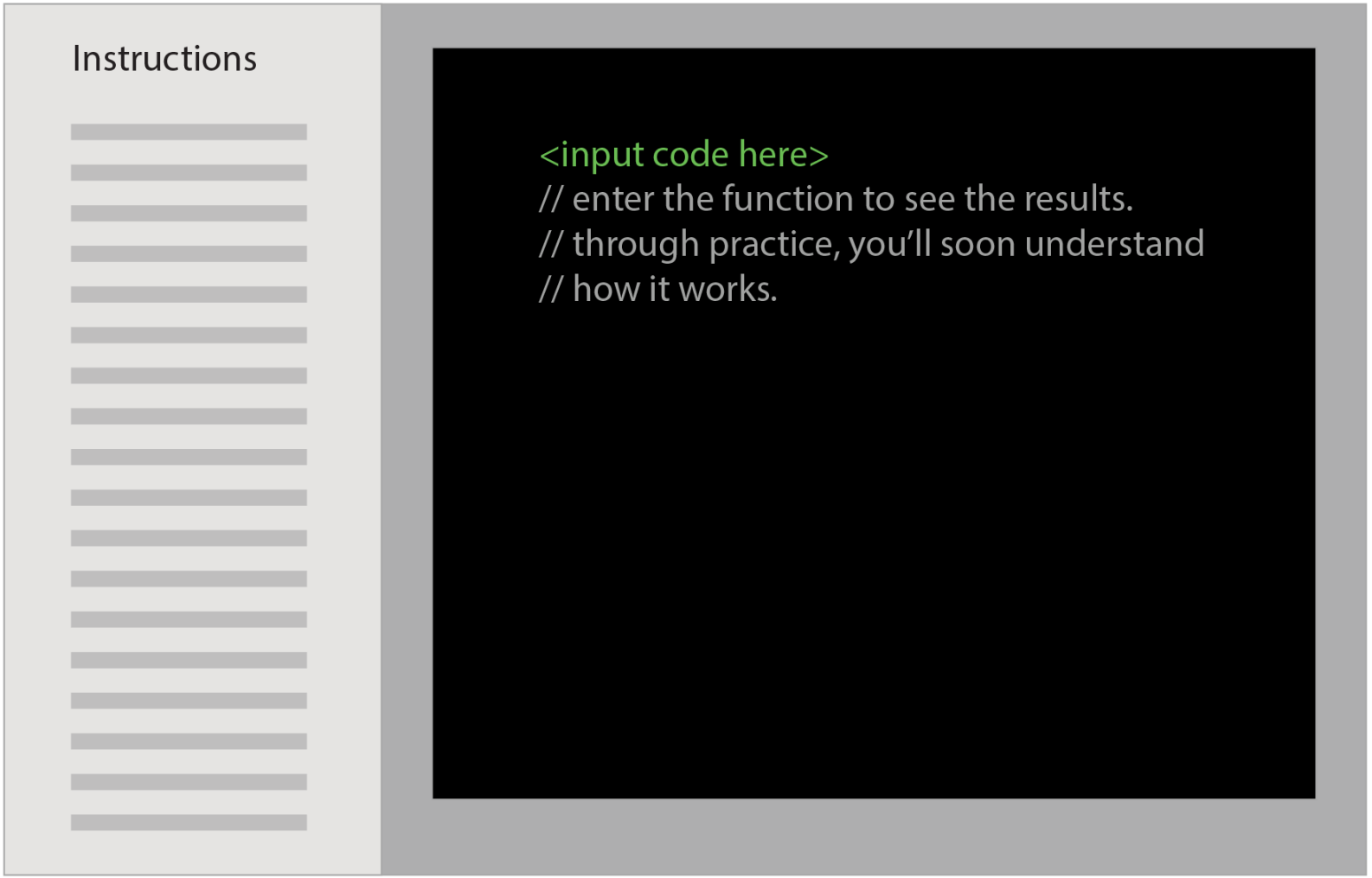
A good search and browse system won't solve findability if the content isn't there. Much of the problem of findability is lack of content itself. Consider writing for the long tail as well as including "known limitations" information.



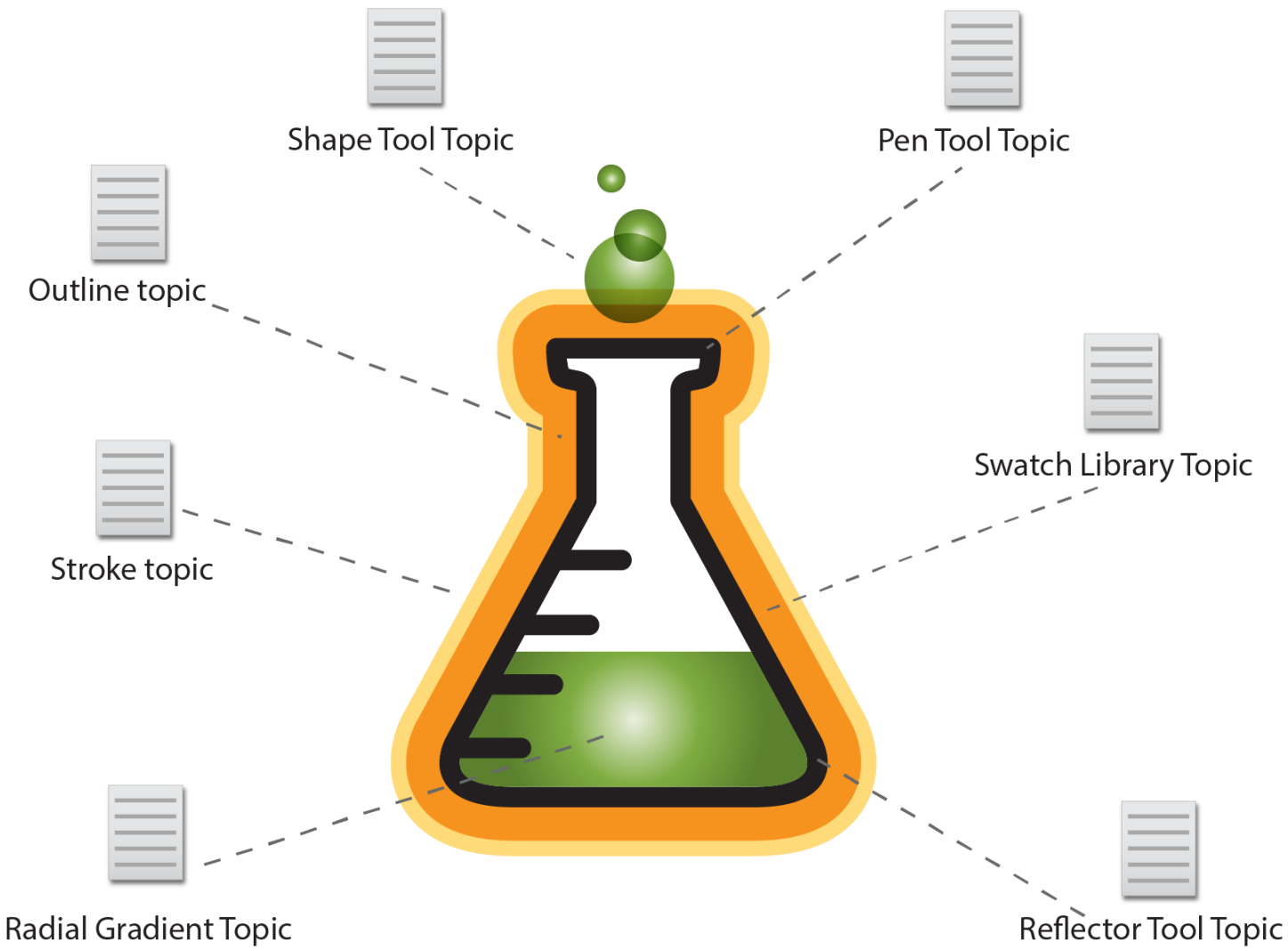
With level-based content, a user needs to master lower-level skills before moving on to more advanced concepts. If you present the user with advanced concepts from the start, the concepts may not mean anything or be understandable to the user, even if he or she finds them.



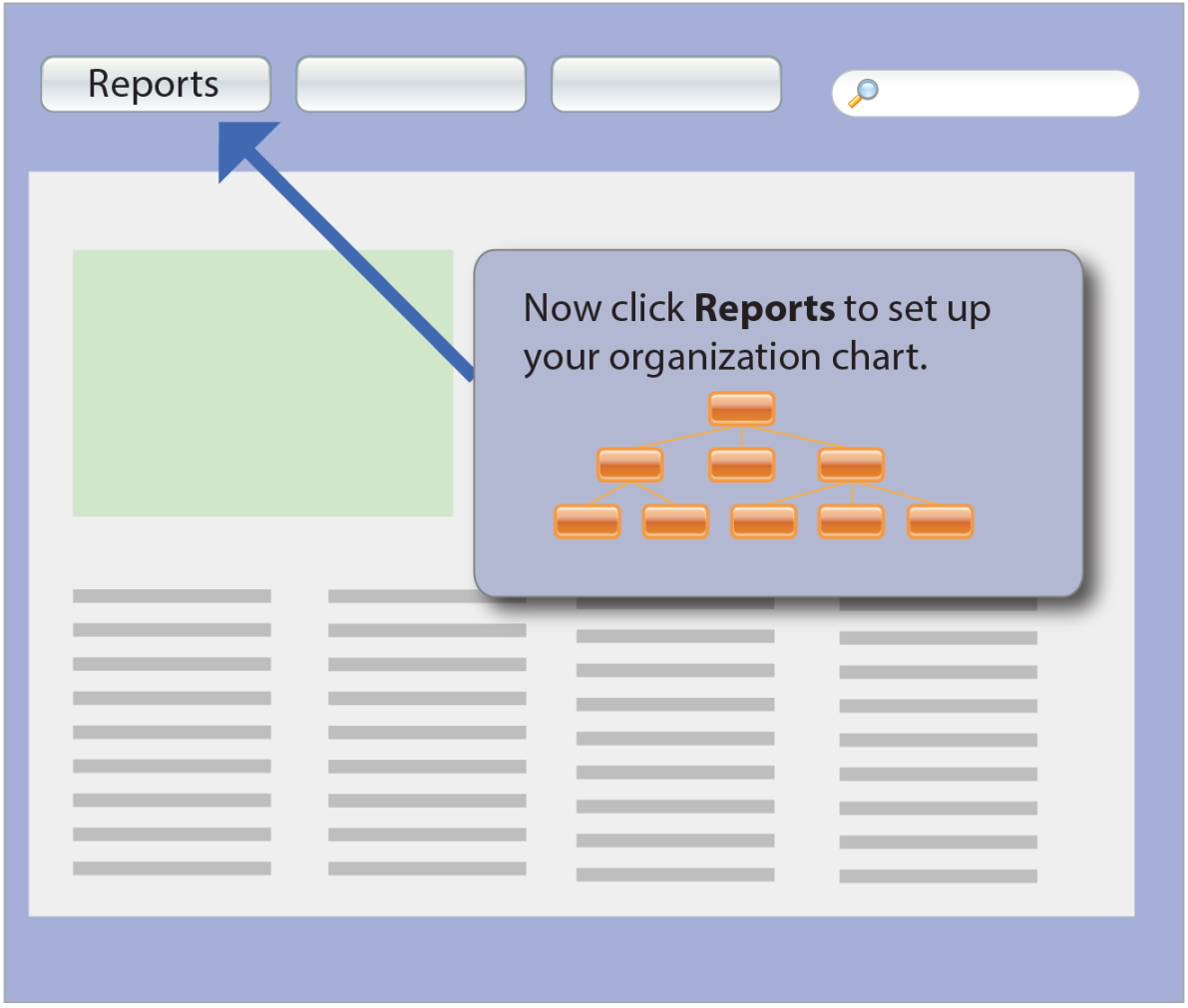
For some users, the knowledge they're looking for isn't consumable on a web page. The users have to realize the knowledge for themselves by acting. It's the "learn by doing" model (see also "Minimalism").



Some knowledge resists organization in a TOC because it relies on a specific combination of a variety of topics. An end-to-end approach helps users understand what can't be conveyed through a single topic approach.



Some users will never find answers to their questions because they never enter the help system at all. They resist looking for any "help" information at all. For these users, you can build help directly into the application.

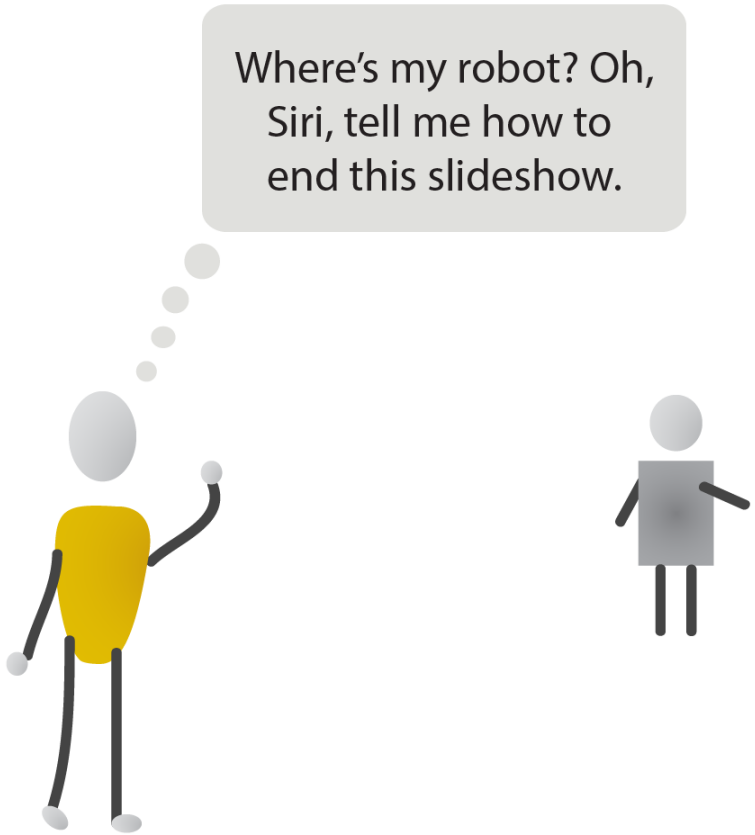


Visual communication helps users understand without knowing specific terms. Visuals can transcend text. Users see meaning without having to connect the terms on the page with an abstract visual model of what those words mean.

text, text, text, text, text, text, text,
text, text, text, text, text, text
text, text, text, text, text
text, text, text, text, text
text, text, text, text
text, text, text, text

text, text
text, text
text, text
text, text
text, text
text, text
text, text, text, text, text, text
text, text, text, text, text, text, text
text, text, text, text, text, text, text, text

Looking ahead 15 years, artificial intelligence (AI) might provide a suite of other approaches to help. For example, how would you optimize findability through audio and natural language processing?



Tom Johnson

- blog: idratherbewriting.com
- twitter: @tomjohnson
- email: tom@idratherbewriting
- phone: 801-822-2241