



# Content Analytics

Moving from Cost to Value

# How and What We Measure

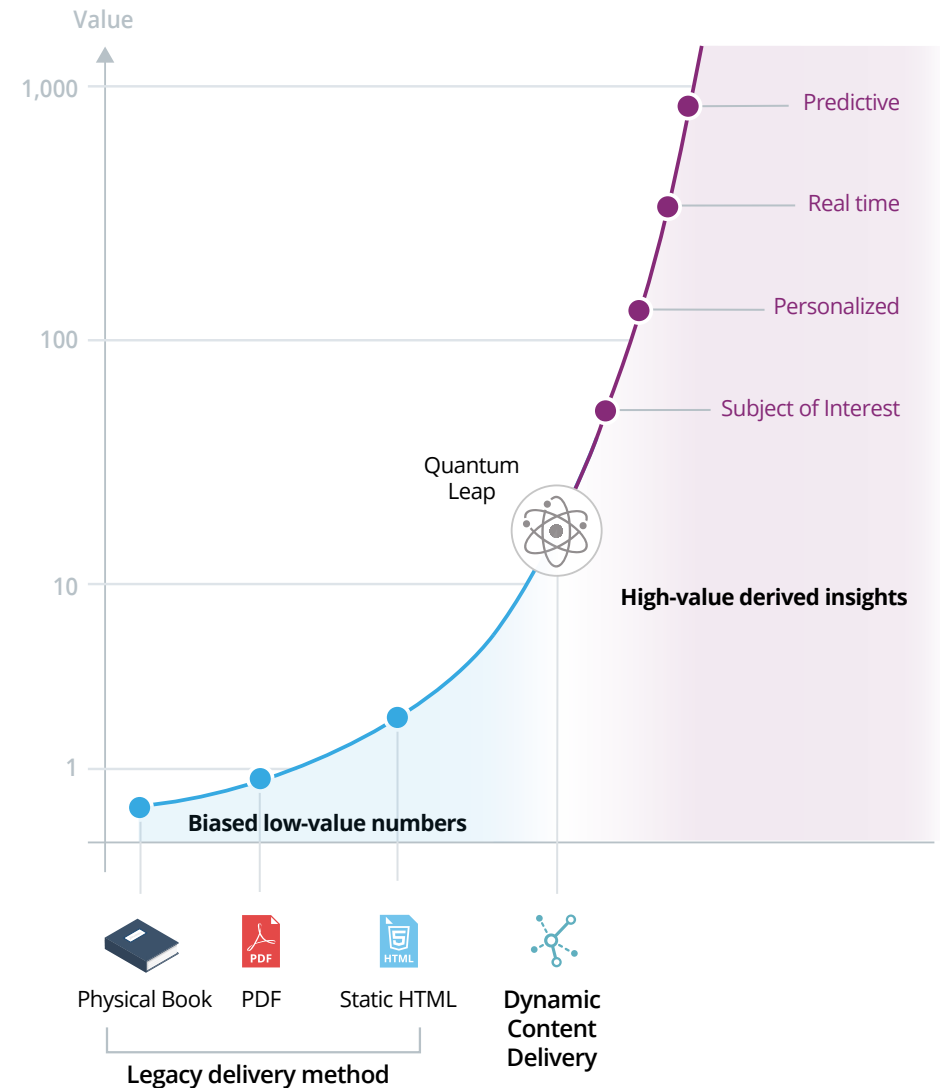
When your tech docs team writes a manual, they print it out, put a nice cover on it, bind it, and slap it down on your desk. Congratulations. That is evidence that they have worked! You've paid for something, and there it is. In other words, that beautifully composed thing on your desk is evidence of one thing: money spent.

Presumably, it has value, or you would never have asked for it. But how do you know the value? How do you measure it? Yes, Dr. Deming famously said, "It is wrong to suppose that if you can't measure it, you can't manage it – a costly myth."<sup>1</sup> but if you have a choice, being able to measure is better. Indeed, we'd rather trust Lord Kelvin when he says "If you cannot measure it, you cannot improve it."

Without measuring the value of technical documentation, all management knows is its costs (all those paychecks and supplies) down to the penny, but the value is taken on faith. It must be worth something, right? Or we wouldn't do it. Let's unpack this.

First, a True Statement:  
*How you measure depends on  
what you're measuring.*

1. John Hunter, The Deming Institute, August 29, 2013  
(<https://blog.deming.org/2013/08/unknown-and-unknowable-data/>)



# How and What We Measure



## Physical Books

If all you do is print physical books, all you know is how many books are sitting in crates. And, guess what? It will be similar to how many customers you have and to how many products you have sold. But you already knew that number (presumably), so not a lot of value has been added.

Of course, we like books. The interface is intuitive, they sit on our shelves reminding us of the good old days, and they smell nice. It's sad to see them go. But they're expensive to print... all those dead trees.



## PDFs

With PDFs, we start dematerializing the books. We convert our manuals into digital files, and post them on a website. Nice move! You have eliminated the cost of printing and shipping... or, at least, transferred it to the customer, at their option. They can always read on their computer (unless they're using the manual to repair the computer... then, the paper, toner, time, trouble, and staples are on them).

You can count how many PDFs are downloaded. After that, nothing is known. Were they read at all? Were they shared on a corporate server with dozens of people? What parts were read? All of that is unknowable. All you get from download numbers is something to let a manager feel that they know what's going on and that the cost of producing the PDF and its content has not been wasted.

This is just not enough data, and not the right data. So, we move on to HTML...

## The Value of the Manual

The old-fashioned way to measure the value of something is to put a price on it and make it available in the market. If people pay for it, it has value.

But if you give it away, how can you tell? Thank-you notes? Maybe you would get complaints if you don't include it. But you have to include a manual, right?

Of course, some companies don't. Apple started this trend in 1999, when it decided not to create a manual for **Mac OS 9**. David Pogue, prominent tech blogger and former New York Times tech columnist, jumped in and wrote *Mac OS 9: The Missing Manual*<sup>1</sup>, which came out in 1999; O'Reilly Media picked it up in 2000. David Pogue has gone on to write hundreds of books, many of them best-sellers, just to fill in this gap between the known cost of technical documentation and its assumed value. O'Reilly has gone on to print many more *The Missing Manual*<sup>2</sup> books; their slogan is, "The book that should have been in the box". They know the value of the franchise via indisputable sales numbers.

So, someone knows the value of Technical Documentation. If you give it away with your product, though, as in a user's guide or reference manual, it is harder to measure. Especially since most software these days doesn't come in a box, and product information can be buried in an arcane file system.

1. [shop.oreilly.com/product/9781565928572.do](http://shop.oreilly.com/product/9781565928572.do)

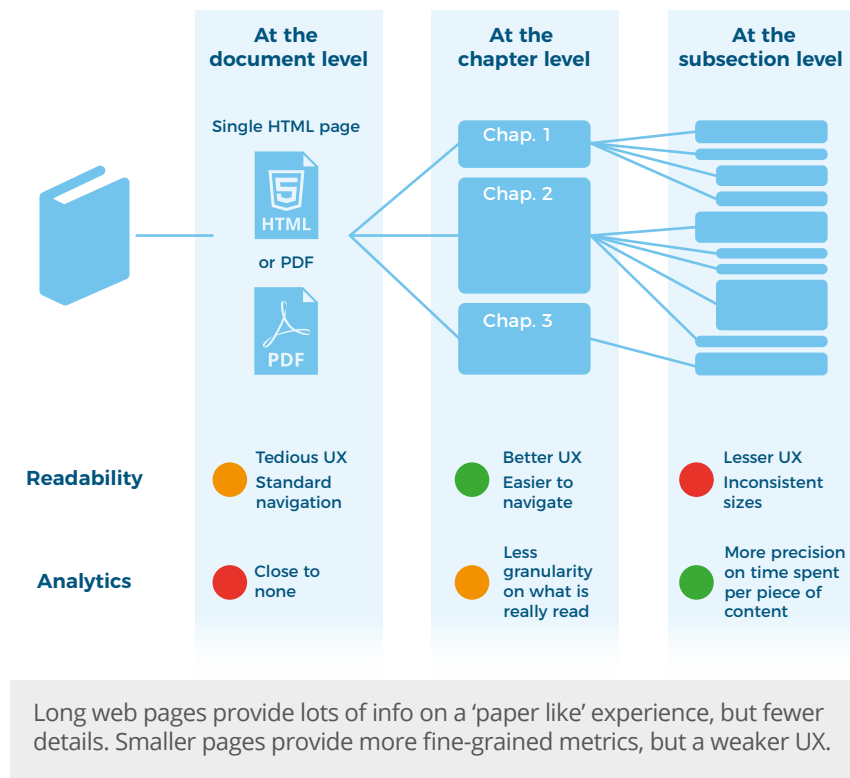
2. [www.oreilly.com/missingmanuals/](http://www.oreilly.com/missingmanuals/)

# How and What We Measure



## HTML web pages

This is actually an improvement on PDFs. First, you know how many times a page has been loaded, when and how long until it's clicked away from. If that time is long enough, it has probably been read. And, since the manual is probably not on a single web page, you have a somewhat finer grain of knowledge. Instead of just being aware that a PDF has been downloaded, you find out that the HTML page containing Chapter 5 of the manual has been loaded into someone's browser. This is better! You can tell management that Chapter 5 has gotten some attention.



If you could tell where on the page the customer was looking, that would be much more informative. Sadly, you (probably) don't own the browser, so you can't get that kind of detailed feedback. If you broke the chapter into many small web pages, you would learn with more precision what parts of the chapter have been read, but it would be a pain for your users: they would spend their time clicking *Next* and *Back* buttons to navigate your content. **So you get to pick – good user experience or fine-grained metrics.**

But what metrics are we talking about? Web pages have been around for a while, and people think they have useful analytics on them using **web analytics tools**, whether Google's or otherwise. After all, they may well use a product called Google Analytics. It's right there in the name! Not to mention that trusted brand. And because they have a lot of data, they may think they know something useful. There should be a name for that – **the data delusion**. Here's why they are deluded.

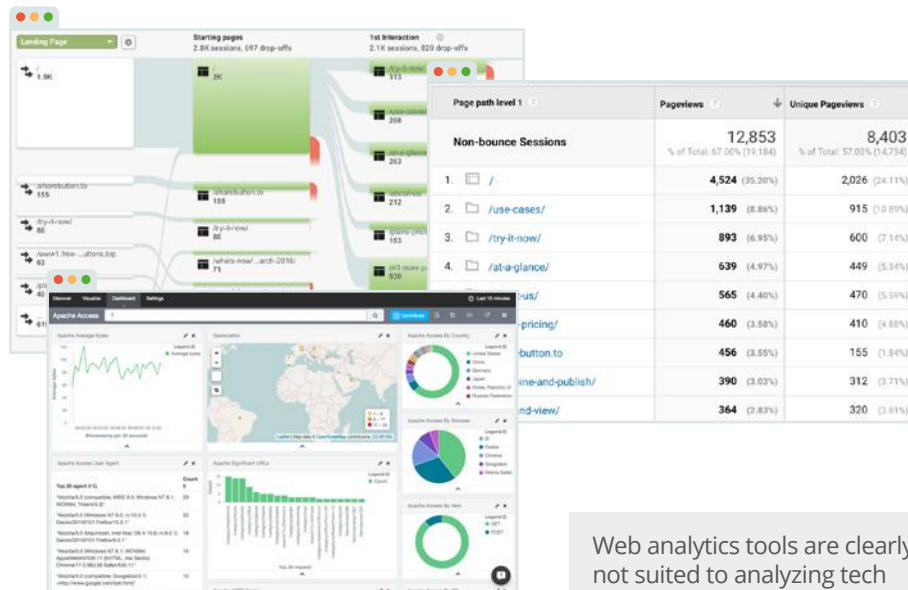
Web analytics products were designed for marketing. It's right there in the summary of their site: *"Google Analytics lets you measure your advertising ROI as well as track your Flash, video, and social networking sites and applications."* Very useful for a company website of 20 to 200 pages. You get a report with the URLs, the number of times they have been loaded, how users got there, where they came from, and where they clicked to next. If you have a small number of pages, with a lot of viewers for each, and you want to focus your marketing dollars, this is very useful.

How many techdocs web pages do you have? Hundreds? Thousands? Given a range of products, each with a set of manuals, in various languages and versions, the number of HTML pages could explode. Not only that, but many pages may only have a small number of readers.

# How and What We Measure

Your report will be a spreadsheet with thousands of items with impossible-to-decipher names and a wide range of numbers next to them. How can you make sense of these numbers without seeing the content at the same time? How can you make sense of a flow diagram showing the path users follow when thousands of small elements must be displayed? In addition, if two pages are related to the same task for the same product, but in two different versions, we need to see the metrics either separately or aggregated; but since Web analytics tools don't understand the structure and semantics of tech doc, it's impossible to automatically create and navigate clustered numbers (by book, by version, by product).

Not to mention that changing the granularity of an HTML page (such as from topic to section) would render any comparison impossible.



Web analytics tools are clearly not suited to analyzing tech content usage.

What kind of action can you take with web analytics information? Not much. Metrics need context to be interpreted; we need to see the numbers and the content together in order to gain insight. Also, web analytics may tell you what has been looked at, but there's a vital bit of info they don't give you – what has not been read. Knowing what's happening at the web server level is not very useful. This renders marketing-style web analytics useless.

So now we know that counting books isn't useful, counting downloads doesn't teach you anything, counting page loads is slightly more helpful, but not much... We need to know what's happening at the user level. **We need a different way to track and log.**

## What we need

If you really want data on how valuable your technical documentation is, you'd practically need to be standing over the user's shoulder, observing what they're reading and for how long.

Crazy, right? Well, maybe not.

We're in the era of the data-driven enterprise, where we can engage the flood of information produced by users. Social media, online retail, mass surveillance, census, all produce torrents of data. Algorithms are grown that can process that data to produce insight. If we could **make our technical documentation generate data** on their use, and apply these advanced big data mining technologies, we could start to truly **understand** how our users are reading them, and **learn** not only more about our techdocs, but about **how our users interface with our products**.

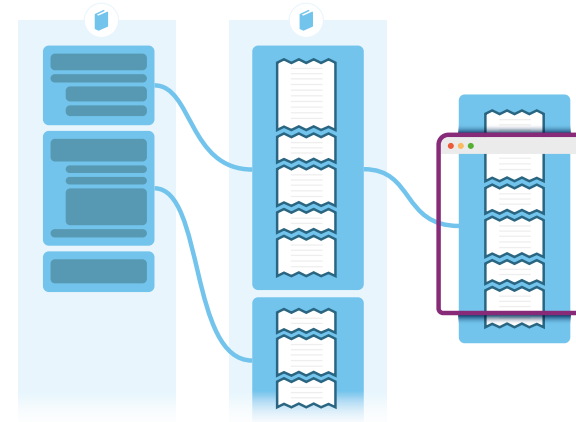
If only a single topic in a manual could send a message home, "Hey, I'm being read now!"

For this to happen, we need two things.



# How and What We Measure

**First, our content needs to be granular**, so that we can track the consumption of each fragment individually. Structured documentation (such as DITA) helps with this, but otherwise, it is the job of the dynamic delivery platform to break the content into chunks, whatever its source format. It must take differently-formatted inputs and make them all consistent and uniform so that you can present the documentation any way you like. Because everything that could be considered documentation – wikis, knowledge bases, user forums, trouble tickets, catalogs, as well as good old manuals and guides – should be searchable through the same portal. It's all product information, and it all contributes to customer success. So it all shall be tracked and analyzed with the same tool.



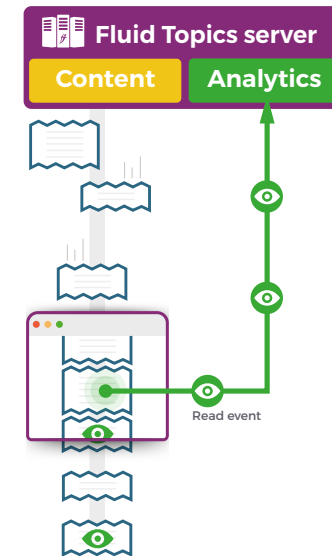
Displaying in a way that hides its fragmented nature creates a good UX while still allowing fine-grained control.

## **Second, we need a custom-designed reading and tracking technology.**

Since you can't control the browser (you don't own it), have your customers view the documentation through a web-based reader that you do control and that sends these messages home when it displays a fragment for a long enough time that it could be read. Tracking the display at the device level, instead of the download at the server level, creates more relevant data.

This way, the benefits of a dynamic delivery portal go in both directions. Your customers read your material benefiting from a contextual and tailored documentation, and you read their access. You get detailed, unbiased data on everything they see, click on, and read, generated on the fly.

By accessing your documentation, your customers are telling you what's on their mind. It's time you listened.



Displaying topics individually allows to track what is really being read and when.



# Reading the Numbers



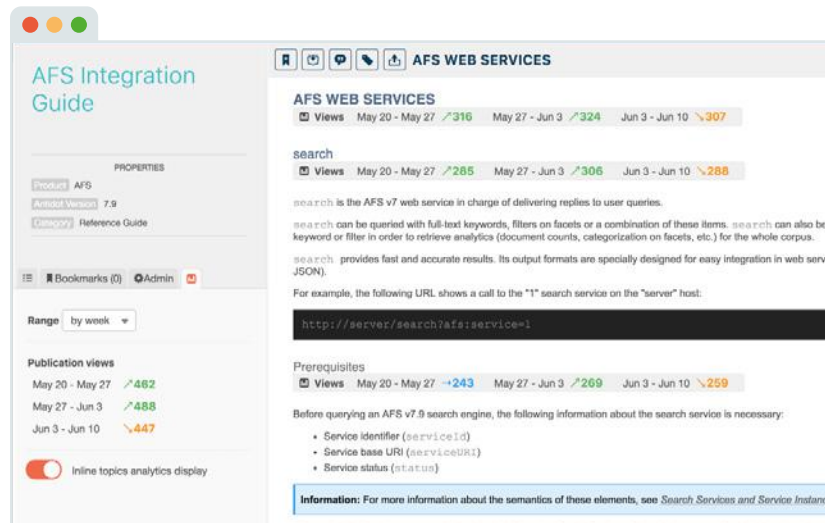
# Reading the Numbers

## Numbers with context

Having done the above, your tech doc team can get a much better grip on what bits of documentation are getting attention and which are not.

The first major improvement we can bring is to display the metrics within the content itself so that you can easily make sense of the numbers by seeing what information they relate to.

Embedding the analytics inside the content is one of the benefits of dynamic content delivery: because the rendering is built on the fly, it becomes easy to enrich the pages.

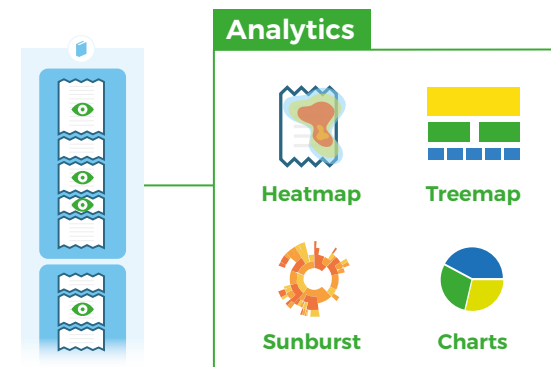


## Data visualization

Using more sophisticated visual representation, you will be able to see the entirety of the documentation using graphical tools (rather than a list of numbers and indecipherable long URLs) and see where readers are focused, where their attention is. What interests them. What they need. And you can do that while seeing the content at the same time, which provides context.

You can aggregate these numbers to get more meaning. You may have an installation guide for each version of your software, but wouldn't it be more interesting to find out how much trouble users are having with installation as a whole, regardless of version?

Metadata are not only useful for users at search time, but they can also be used to create clusters and axes for exploring the numbers. See content consumption per product line, type of content, etc. Possibilities are endless, and creating aggregates dynamically helps you understand and reveal the hidden data.



Using sophisticated data visualization such as heat maps, sunbursts, tree maps, and others, you can gain immediate insight into what is driving your users.



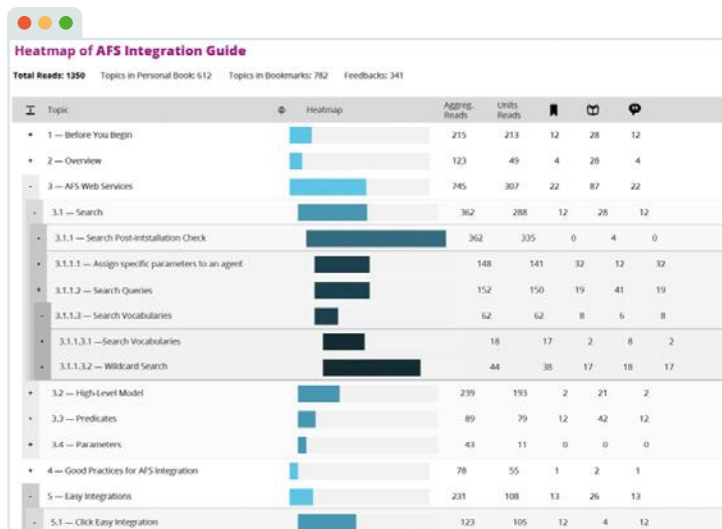
# Reading the Numbers

## Taking action

Maybe your users need to figure out how to set the clock on the stove. If your technical writers look at the data and see that this is a popular topic, maybe they could shoot an email over to Customer Support, and give them a heads-up that this is bothering users. Or they could forward reports to Product Design and let them know that their interface is confusing – maybe they can do something about that in future models. Or they could contact Sales and suggest that when they sell an item, they chat about how to set the clock.

The possibilities are suddenly blossoming because you have discovered the data hidden in your documentation consumption.

Your technical writers have gone from being a known cost with an unknown value to being a known cost with a known value. Your data-driven organization now has a data-driven techdocs group.



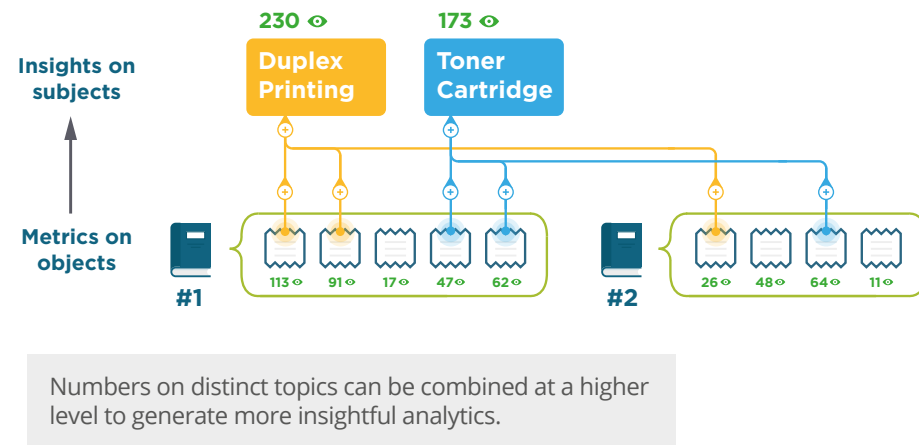
# A New World of Possibilities

With data flowing out of your dynamic delivery portal, you can do many more things than you could before.

Once you are able to collect data on every topic your users read, when and for how long, you can create a data portrait of each of them. Your software can start to group them in novel ways that people, full of biases and preconceived notions, could not.

## Subjects of interest

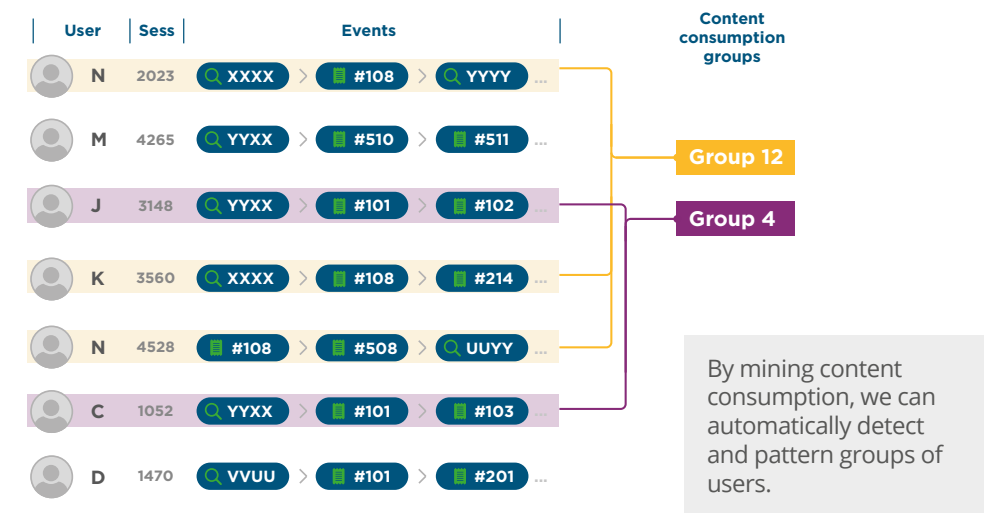
Beyond visualizing the consumption of content fragments, you can begin to perceive what subjects interest your users, not just which bits of documentation they are looking at. There will be forum posts, trouble tickets, knowledge base entries, and, sure, user guide topics, all having to do with **the same subject**, “How to restart the XY module” for instance. The combination of advanced text-mining and data-analysis algorithms permits you to see the bigger picture – not what documents your users are looking at, but what problems are bothering them.



Now we have information that is not just interesting to tech doc, but to engineering, sales, management, product design, and, indeed, the entire enterprise. If calibrating the power supply is a persistent pain point to your field techs, maybe your support team needs to get in touch with engineering to figure out how to offer better support; your product design team might want to reconfigure it in the next version; and your training team could devote extra time to that topic.

## Patterns and personalization

Once you know what subjects attract your readers, you will start to notice clusters of users, all interested in the same subjects. This gives you unprecedented power to offer suggestions based on what users with a similar reading pattern have looked at. You can offer your users a kind of **curated serendipity**, by using similar users' searches to affect result ranking.



# A New World of Possibilities

Search results can be tailored to the individual. If a user is always looking at expert maintenance content, what is the point in proposing basic installation documentation? Personalization is even more important when the delivery channel is not a web browser, but more constrained devices such as heads-up displays and chatbots.

| User Group | Content |        |        |        |          |
|------------|---------|--------|--------|--------|----------|
|            | Book 1  | Book 2 | Book 3 | Book 4 | Art. 508 |
| 1          |         |        | ✓      |        |          |
| 4          | ✓       | ✓      |        | ✓      | ✓        |
| 12         |         |        |        |        | ✓        |
| 274        |         |        | ✓      |        |          |

Based on user C search history, promote content preferred by group 4

Search engine results and suggestions can be based on user characteristics.

## Real-time support

Using the new power of being able to hear and see your customers in the act of looking at your documentation, you can offer targeted, smart, real-time support. Instead of giving your chat window a dumb rule, like “pop up after two minutes,” you can give it a smart heuristic, like “pop up if they’re looking for subject X, with a relevant suggestion.” Suddenly you’re being helpful instead of covering their browser tab with a useless interruption. And if your phone support agents can see what the user has been searching for, they could be more helpful, faster than ever before.

No more will they have to ask an expert user if they’ve tried rebooting, or offer a novice a complicated download and install. They will know the data-portrait of the user and adapt to them.



Popup windows could be displayed at the right moment by tracking user behavior and content consumption.

## Predictive support

From looking at the past with data visualization, to perceiving the present with real-time support, you can look at the future. If a certain pattern of search has tended to result in a call to support, you can start to predict when such a call is likely, and reach out to the customer first. A quick easy call now could prevent a difficult, lengthy call later.

# From Cost to Value: Conclusion

Your technical documentation started as something you sent out in the mail. All you knew is that it cost something to create and ship, and that you had to do it. Now it has transformed into a medium of communication between you and your users. And it's even better than a conversation, because it's totally honest. Your user is interacting with the portal and has no reason to be anything except totally candid.

Do it right, and it can tell you what's bothering your users, and what you can do to help.



# ABOUT THIS GUIDE

## Content Delivery Analytics: Moving from Cost to Value

This white paper explores how you can turn tech content into a sensor and its delivery into a data generator. Understanding how product documentation is consumed can fuel your company with data that has the potential to transform operations and impact strategic decisions. To gain this insight, you need to change the way you track and mine the behavior of users when they search, read and interact with your technical content.

### Table of contents

#### CONTENT ANALYTICS

How and What We Measure ..... 2

#### READING THE NUMBERS

Reading the Numbers ..... 8

A new world of possibilities..... 10

From Cost to Value: Conclusion ..... 12

### What's behind Fluid Topics

Fluid Topics is a perfect name for a dynamic publishing product.

"Fluid" conjures images of effortless motion, smooth and easy flow, and graceful simplicity in movement or execution.

"Topics" evokes conversation, communication, learning, thinking, reasoning, education, and ideas.

Together, **Fluid Topics** conveys a smooth action or movement of reaching for and shaping information, quickly and effortlessly without limit or barriers.

**That is what we do.**

### Contact us:

**For technical questions:**

*[fluid.support@fluidtopics.com](mailto:fluid.support@fluidtopics.com)*

**Website:**

*[www.fluidtopics.com](http://www.fluidtopics.com)*

**Blog:**

*[www.fluidtopics.com/blog](http://www.fluidtopics.com/blog)*

**Twitter:**

*[@fluidtopics](https://twitter.com/fluidtopics)*

# Content Analytics

Moving from Cost to Value



[www.fluidtopics.com](http://www.fluidtopics.com)  
[info@fluidtopics.com](mailto:info@fluidtopics.com)

Revision: August 2018  
© 2018 Antidot, All rights reserved.

Fluid  Topics