

# A strategy to address the findability problem

As I mentioned in a recent post, poor findability is often found at the top of the list of customer experience issues. In fact, at Red Gate it was right at the top of our to-do list for improving the customer experience. Here's how we approach solving the problem...

## Findability improvement strategy

We've been working on the findability problem for a while at Red Gate, starting in Technical Communications and Support (with the information that people need to learn how to use a product and to troubleshoot any issues), and then moving out to more general online customer experience, and we've identified that really addressing this problem needs continuous improvement.

## Why findability needs a continuous-improvement approach

**1. Poor findability is a complex problem, involving interaction of many elements** (content, search engine... [see this previous article for a full list](#)), and the relationship between elements isn't always predictable. This makes it impractical to implement a big once-and-for-all solution.

**For example:** the following factors are likely to improve the ability for people to find what they're looking for via a search:

- change the terminology in existing content
- write new content that matches common search terms more closely
- improve page titles
- change the layout of search results
- Introduce new features such as auto-suggesting search terms or introducing faceted search to help the visitor filter results down to the right product, date or other category.

In an ideal world it might be great to do all of these ... but in a more realistic world, we need to think carefully about what we're spending time on, so we're likely to only a selection of these. This gets good enough results, providing you choose the right selection, and you don't waste time and resources on unnecessary work. Since the interaction between these different factors is difficult to predict, it makes sense to introduce them one or two at a time, and monitor improvements until you reach an acceptable level of findability.

**2. Findability isn't something you can fix and then forget.** As long as you are adding new content to your site and/or site visitors have new expectations from you (e.g. based on a product your competitor has added to their site), the context people are searching in is continually changing.

**A simple example of this:** for a long time we had a product that dealt with backing up SQL Server databases; one of its features was the ability to restore from a backup. When visitors entered "restore" as their search term (yes, they do enter such vague search terms!), we could make a good guess that they wanted to find out something about our backup product.

Recently, we introduced a new product, also with a "restore" feature. Now when people enter the search term "restore" we're not so sure what they're looking for.

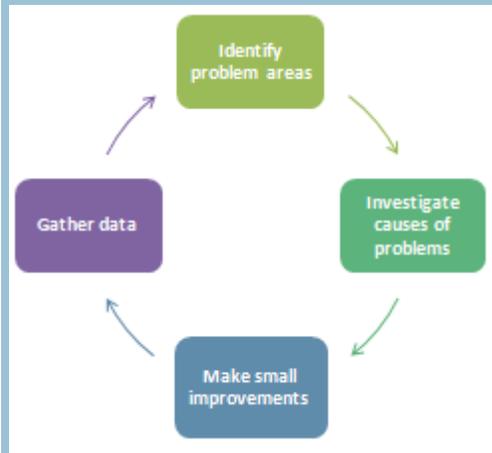
The original product documentation hasn't changed at all; neither has the site visitors' information-seeking behaviour. But the addition of a new product has completely changed the context that visitors are searching in.

**Here's another example:** 2 years ago we implemented our "support centre", bringing together product help, tutorials and knowledge base in one part of the site. We designed and tested our information architecture, and it was adequate for needs. However, as part of this project we'd also made it easier for the support team to publish knowledge base articles online, and they seized this opportunity with gusto. Within just a few months, the number of KB articles had doubled. Our growth predictions for content on this part of the site hadn't anticipated this, and the navigation wasn't up to the job at all – we presented site visitors

with long lists of 40 or 50 articles, where we'd anticipated just 10-20; and this made it very difficult for them to spot useful content.

## Improving findability

Our strategy for improving findability is basically very simple:



### Gathering data and using it to investigate problems

We tend to use quantitative methods to identify likely problem areas, and then qualitative data (and sometimes additional investigations too) to investigate the causes of problem.

Across the business, we monitor a range of data:

- Quantitative data about customer contacts (e.g. from support logs) – number of calls, how long it takes to resolve issues, how much to-ing and fro-ing to resolve
- Qualitative data about customer contacts (e.g. from support logs) – what are the calls about, does content exist that would answer the question
- Customer satisfaction & NetPromoter scores
- Feedback on customer satisfaction scores
- Feedback on web poll / survey
- Search data, from web analytics\* – what terms are people searching for, what pages do they look at, what searches have high exit and refinement rates
- Search data from precision and relevancy tests\* – how good is our search at returning relevant results
- Navigation data, from web analytics – where do people arrive at the site from, how are they interacting with specific pages (how long do they stay on the page, what do they do there...)

### Making improvements

From a list of potential improvements, we identify a small number of high priority ones to implement (rather than implementing everything; see the first “continuous improvement” bullet point above for reasons). This gives us the ability to respond and understand success – or not! – fairly quickly, and then run through the cycle again.

### Does this strategy work?

We followed this strategy for some time within the technical communication parts of the business, developing an understanding of what help or troubleshooting content really needs to be written for a specific

product, based on support calls, search terms, and so on. ([Take a look at this previous article for a bit of information about our early explorations into using data to evaluate this type of content.](#)) We've had success here in cutting down the amount of content we deliver alongside releases – because we know we'll be able to identify any needs that emerge quickly and address the problems.

We've also begun to use this recently as a way of implementing and tweaking our new site search. We're still in the early days of this one (just near the end of our 2nd cycle), so we're the additional requirements emerging are often fairly large (e.g. new search features).

### **Some implementation “challenges”**

Getting success with this strategy depends on separating content-writing cycles from other business activities – such as developing and releasing new versions of products.

Our cycles aim to be 3-months long, and early cycles following a release tend to take more work to analyse and make improvements, whereas in later cycles there's typically less that needs doing.

With both of these examples, what we've found is that it really relies on collating data from across the business, and making sure that we have the ability to judge the success and value of improvements. It's fairly early days with getting this working, so we're still disentangling some of the issues... I'll let you know how that goes!

Source: <http://communicationcloud.wordpress.com/2010/07/30/a-strategy-to-address-the-findability-problem/>