# Writing the Right Stuff

We have a lot to document and a lot of people with an interest in what gets documented. That's why I'm always talking about prioritization and alignment: choosing what to do, and getting people on the same page as you.

In this note, I'm going to go over what we do and who we do it for. I'm going to talk about:

1. Users
2. Stakeholders
3. Quality

## Users

We're technical writers. Of course we're going to advocate for the user. Who is the user?

As you know, our users are developers who build data integration pipelines. Sometimes prioritization is as simple as documenting what's most important to them. But that's not always as straightforward as it sounds!

Here's a different question: Where do our users come from?

Some of our users are migrating from the legacy version of the product, and some are not. These groups have different needs and different relationships with our stakeholders, the people we generally work with to get input on the docs.

## Stakeholders

I'm going to talk about four groups of stakeholders who we work with pretty closely: development, product management, internal users, and design.

## Development

I think we're pretty lucky to have a great dev team, since we work with them more than anyone else. Our developers do a lot to help us ensure the technical accuracy of our documentation. That being said, their priorities differ from ours in some important ways.

*Feature focus*
Naturally, when you build something, you're going to want to tell people all about it.

Often developers will give us all the details they can think of about a feature. It's up to us to identify information that will actually help someone use it. Emphasizing task orientation can help with this.

*Assumption of expertise*
On the other hand, developers can be so familiar with what they're working on that they forget to mention details a user wouldn't know.

This problem is compounded by the legacy/new user split. Legacy users are often working for big customers who can directly contact our dev leads to ask for features or documentation. A developer's main context for a feature might be "Customer X asked for it," so from their perspective it's going to be used by someone who already knows exactly what it's for.

As writers, we can balance this out by keeping new users in mind. We investigate and document context to ensure that features are accessible to any user who needs them.

### Product management

Our PMs are always thinking about how to appeal to new users, and they can provide great support for our work to create a cohesive doc experience. It's their job to see the big picture.

That helps a lot when we're making the strategic case for building out an information architecture. Since we're always getting a hodgepodge of requests from legacy customers, the overall usability of the docs suffers if we're too quick to ship. Coordinating planning with PM helps the docs stay well-organized even when we're producing everything everywhere all at once.

### Internal users

Our internal users include sales engineers who demo the product for prospective customers and CSMs who help existing customers succeed. Some of the most critical tests of our documentation happen as part of their work in the field.

Understandably, they're often very focused on a specific problem, but collecting information from them over time can be very helpful to understand how well the docs are working for first-time users. They're also a great source of information about user pain points.

Sometimes we get very specific doc requests from internal users. I recommend checking with PM to confirm what information is safe to expose and checking on our own doc agenda to see if we can cover items as part of a bigger plan.

**Design**

I'm including design in this list because even though we don't work with them on much documentation, we do work with them for UX writing. There are two main considerations we have when working with design.

*Embedded assistance*
From a designer's POV, text is often something to minimize. Often we help designers out by providing clearer and more concise restatements of UI text. Sometimes, however, we have content in the documentation that would really be more useful if it were in the UI, and it's our job to advocate for that.

*Terminology*
I've seen a single feature referred to by three different names: one by dev, one by PM, and an entirely different one in the UI. Who knows what the sales folks were calling it? As writers, we protect the usability of the docs when we enforce consistent terminology.

Defining good terminology requires balancing stakeholder needs. It helps to meet people where they're at; this is why we collaborate with PM on blogs and use Github to communicate with dev. It makes it easier to align on how we talk about things. We work with design to define terminology and bring other stakeholders on board, because the name users see in the UI is the one that's going to last.

**Quality**

Sometimes, good documentation is obvious. Sometimes it depends: who are we writing for?

When we write for legacy users, we tend to make life easier for our devs. We encourage design to reuse feature names from the old product. We cover as many migration details as we can. We follow up closely on known issues and limitations and let users know as soon as they've been fixed.

When we write for new users, we tend to make life easier for our PMs. We change terminology to be more useful and intuitive. We include details and context that are helpful for planning. We invest time into organization and architecture.

Sometimes we prioritize quality by pushing back on what we’re asked to do. We push back on documentation requests that would make more vital information harder to find. We push for suitable information to go into tech notes, blogs, and the UI.

Quality is contextual. We have limited resources and we don’t document in a vacuum. So it’s important to be aware of tradeoffs and how they affect the people we work with, as well as our users. When we give a reasoned, empathetic explanation of our choices, we strengthen the stakeholder relationships that help us deliver our best work.