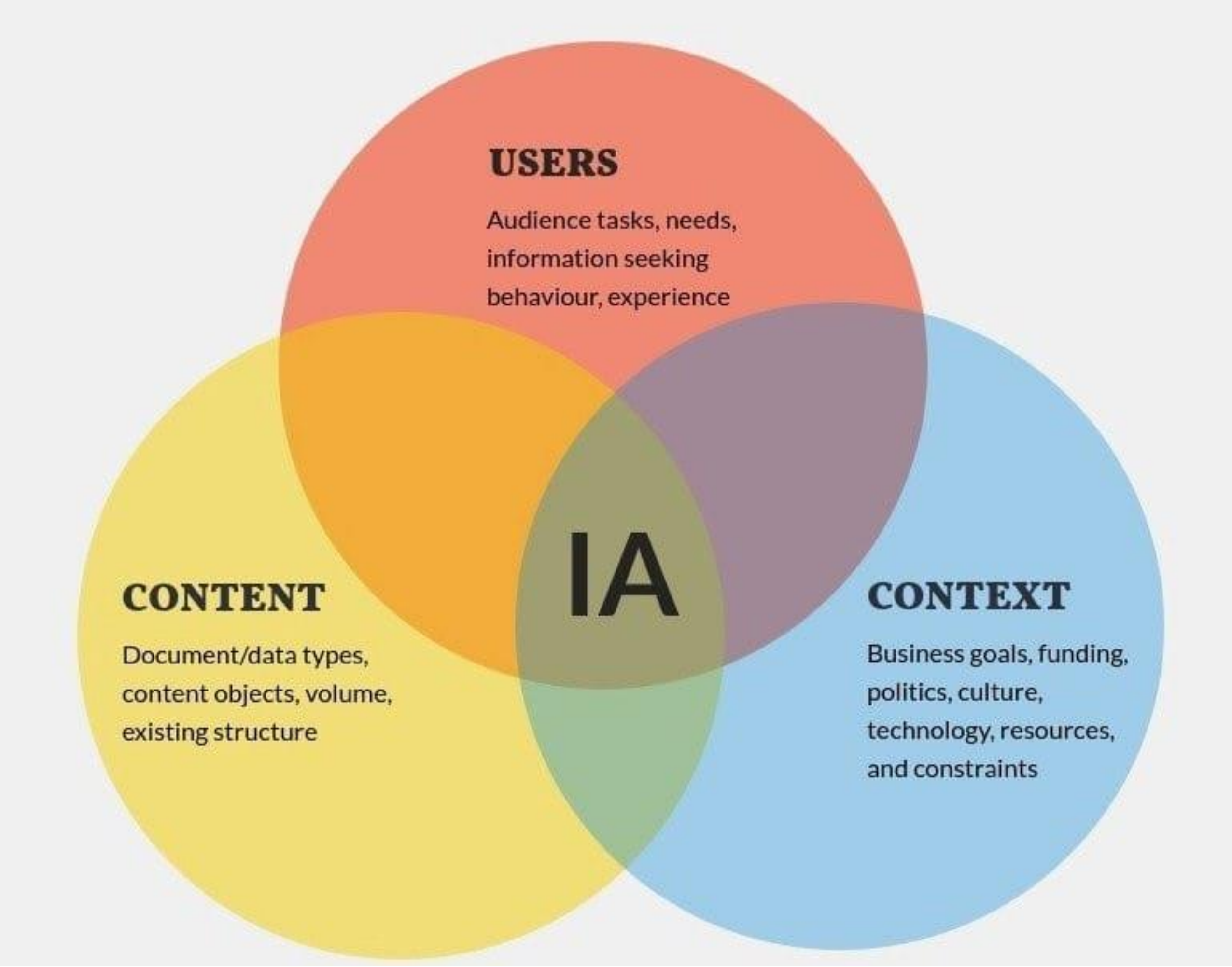
Portfolio of work

Mysti Berry


USERS
Audience tasks, needs, information seeking behaviour, experience
CONTENT
Document/data types, content objects, volume, existing structure
IA
CONTEXT
Business goals, funding, politics, culture, technology, resources, and constraints

Information Architecture Project

My first IA project aimed to stop the “link pain” for the Salesforce technical writing team and improve the customer experience. Our doc team was about 50 people at the time.

The pain

We’d been debating what to do about “the link problem” for a long time, when a rare opportunity came along: a two-week development hiatus!

The link problems were small, daily “paper cuts” on our efficiency:

- Writers had added many cross-reference links to topics without rules or patterns. Moving or reusing content was very time-consuming as there were many link issues to resolve. As a result, writers often chose to avoid restructuring or reuse when it was needed.
- Our resource manager (Arbortext) was difficult to use, taking over a minute to create a single link.
- Customers were often exposed to topics with a distracting number of links or links that created a circular path.

The investigation

During the investigation phase, we discovered:

- Links slow every reader down and decrease comprehension as measured by recall.
- In our highest viewed, lowest rated topics (where you’d expect the most link-clicking), any link on a page was clicked between 0%-8% of the time. Map topics, which exist only to get a customer to another page with a link, had 20%-35% clickthrough, meaning about 70% of the time, people chose to return to search rather than click a link. These statistics were obtained using a small sample. However, informal chats with information architects at VMWare confirmed these results.

The plan

I planned and ran the project for our entire team to take advantage of a rare two-week break in development to fix the problem. I recommended two changes to any topics that wasn’t a map topic:

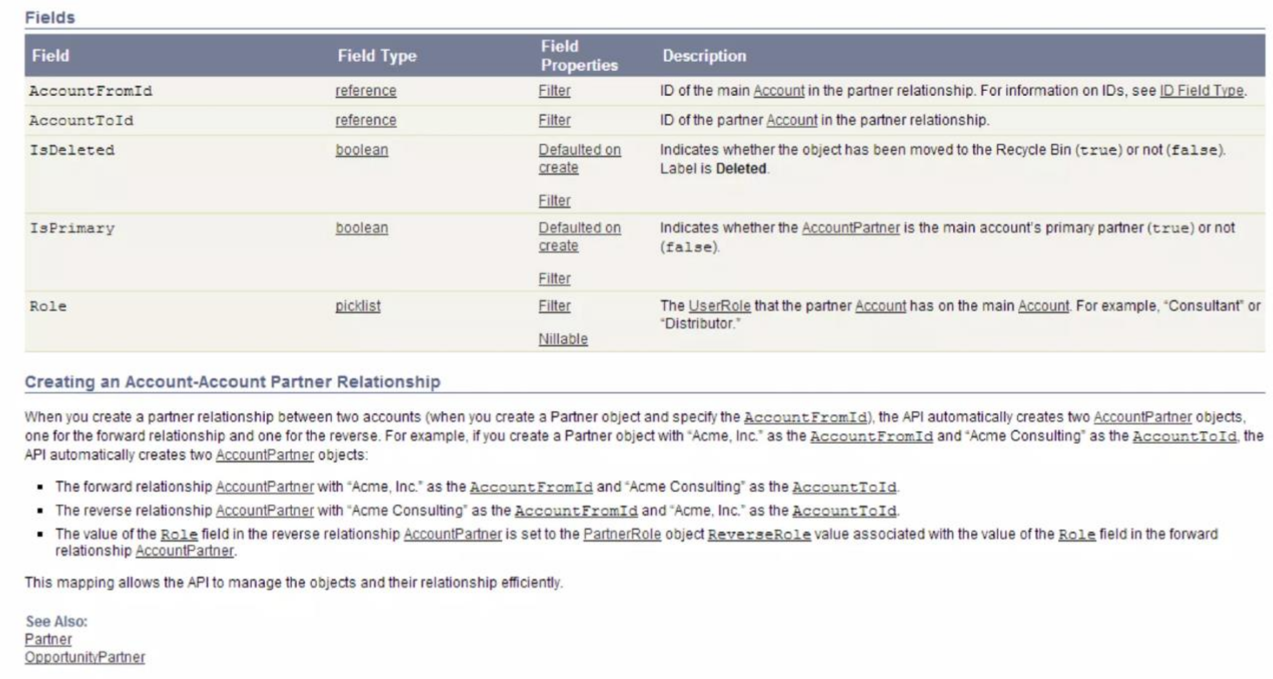
- Our style would be to put all links in a “See Also” section at the bottom of a topic so that links wouldn’t interrupt the topic at hand and wouldn’t interfere with moving content around or reusing it.

- We would only keep links to “must have” content (content that could prevent an error or provide clarity where we knew customers got confused).

We also audited our content to find existing links and evaluate their usefulness. We removed unnecessary links and moved helpful links to See Also sections in dedicated sessions.

The results

Our topics started out looking like this:



Fields

Field	Field Type	Field Properties	Description
`AccountFromId`	reference	Filter	ID of the main Account in the partner relationship. For information on IDs, see ID Field Type.
`AccountToId`	reference	Filter	ID of the partner Account in the partner relationship.
`IsDeleted`	boolean	Defaulted on create Filter	Indicates whether the object has been moved to the Recycle Bin (`true`) or not (`false`). Label is **Deleted**.
`IsPrimary`	boolean	Defaulted on create Filter	Indicates whether the AccountPartner is the main account's primary partner (`true`) or not (`false`).
`Role`	picklist	Filter Nillable	The UserRole that the partner Account has on the main Account. For example, “Consultant” or “Distributor.”

Creating an Account-Account Partner Relationship

When you create a partner relationship between two accounts (when you create a Partner object and specify the `AccountFromId`), the API automatically creates two AccountPartner objects, one for the forward relationship and one for the reverse. For example, if you create a Partner object with “Acme, Inc.” as the `AccountFromId` and “Acme Consulting” as the `AccountToId`, the API automatically creates two AccountPartner objects:

- The forward relationship AccountPartner with “Acme, Inc.” as the `AccountFromId` and “Acme Consulting” as the `AccountToId`.
- The reverse relationship AccountPartner with “Acme Consulting” as the `AccountFromId` and “Acme, Inc.” as the `AccountToId`.
- The value of the `Role` field in the reverse relationship AccountPartner is set to the PartnerRole object `ReverseRole` value associated with the value of the `Role` field in the forward relationship AccountPartner.

This mapping allows the API to manage the objects and their relationship efficiently.

See Also:
Partner
OpportunityPartner

We removed 46% of all “inline” links (links not in map topics or See Also sections), about 4,300. At that time, we had about 4,000 help topics. We didn’t collect metrics on the results. However, writers were more willing to tackle restructuring projects, and some reported satisfaction with how much easier it was to maintain content that wasn’t entangled by links.

The current version of the topic still has no links outside the See Also section, over ten years later.

We retrained writers in the following ways:

- Instead of linking to a help topic that tells the user what part of the UI to visit next, tell them to go to the next part of the UI.

Example:

Old: **Enable CoolNewFeature**<*link to help topic that says "CoolNewFeature is in Setup > Customize">*

New: **Enable CoolNewFeature in Setup: choose Customize > CoolNewFeature**

- If linking to a small amount of text, consider reusing the content instead of linking to it.
- Remember that every link has costs and benefits to the Doc Team and the customer. Weigh the relative costs and benefits before linking.

Writing Samples

I created and edited the following samples.

Usage metering for value-based pricing (VBP)

Value-based pricing has many factors to consider, and pricing is very important to the customer. This document bridges a product gap—eventually, all of this will be displayed in the UI.

Tiered Events

mParticle changed their pricing model to value-based pricing and introduced tiers of events that customers could choose to save money.

Note: Most of the images after the first one were added by someone after I left. However, it looks like most of the text is as I originally wrote it.

API Tutorial

I wrote this from scratch four years ago. It looks like only minor edits have been applied since that time. I broke it up into discrete HTML pages so that we could measure how far people got through the tutorial before giving up.

Note: Images have been updated, but most of the text is as I wrote it, and the structure is the same except I don't remember using sub-steps in Step 5.