Case study: Flare vs. DITA

PRESENTED BY

Jayna Locke
About me

• Content Strategist and Tech Pubs Manager, Digi International
• 3 decades as a communications professional in multiple roles:
  – Technical writer
  – Technical marketing writer
  – Website content developer
  – Marketing communications professional
Quick poll!

• How many of you have evaluated DITA at some point?
• How many have actively used formal DITA?
• How many of you have no idea what DITA is and are only here because the other rooms were full? (Kidding!)
Tech Comm problems in a nutshell
If you have more than one doc....

• You must ask the question: Can we single-source our content?
• You may need to coordinate work between more than one team member.
• You need an established publishing workflow.
• These issues become amplified if you have:
  – Multiple products
  – Multiple writers
  – Translation needs....
Why we began looking for a solution
Our writers were siloed

• In the good old days, our tech writers were only required to be responsible for their own content.

• As a result....
Our problems were legion

• The writers worked in isolation.
• Two writers sitting right next to each other could be employing vastly different methods – voice, style, formatting, tagging, etc.
Single sourcing was isolated

• Writers would often spin off a new document from an existing one. This occurred dozens of times.
• Therefore, we had a giant repository of similar but different content.
• Other than copy/paste, there was no re-use, and no “single source of truth” for duplicate content.
Why that matters

• What happens when you find an error in a document that has been duplicated numerous times? Or when your product is rebranded?
• Without a solid single sourcing methodology, the only option is to go into each of the different documents and fix them all one-by-one-by-one.
To complicate the situation...

- We had grown by acquisition. We were like five companies in one.
- Every company had done things differently, and this meant:
  - Complex product lines
  - Different voice and style across the doc repo
  - Creative use of source control (i.e. sometimes source was missing), and...
Multiple doc tools

• Our source tools included:
  – FrameMaker
  – Microsoft Word
  – InDesign
  – DreamWeaver
  – MadCap Flare
  – Illustrator
And we had tagging Armageddon

Our hundreds of documents all had unique sets of tags:

• H1
• Head1
• Heading1
• Head1-indent-5pt
• Paragraph
• Paragraph-italic
• Paragraph-code….
Formatting was out of control

- Writers could over-format to their hearts content.

- “Oh, you want this section in purple? You got it!”
The reviewers were distracted

- When each document seemed to have its own personality, the developers found it jarring.
- Their attention was on the use of bold and whether code samples were in gray or white boxes, instead of technical integrity.
And so we began our journey

• We hired a consulting team.
• We spent months in a discovery process to quantify the problems.
• We performed a complete content inventory.
• We ran collections of docs through a content re-use analyzer tool*.

* Harmonizer, by DC Labs
Our assessment was revealing

• We identified multiple ways in which our inefficiency was costing us money:
  – Formatting content was highly time consuming.
  – Review processes were outside the workflow. Therefore, writers had to track them down and harangue reviewers for input.
  – Translations were incredibly costly: multiple instances of the same content all had to be translated.
Other costly methods

• We had to maintain licenses for all six of our content development tools.
• Our writers had to master each of these tools, or be completely lost when picking up a new project.
• As mentioned, there was no baseline content. So writers duplicated docs or created new material from scratch instead of single sourcing.
• The team was disenfranchised and frustrated. We had a high turn-over, which meant high recruiting and training costs.
And more costs of inefficiency...

- Support costs:
  - Our Tech Support team would look for troubleshooting answers in our documentation, but would quickly give up because the right information was too hard to find.
  - Instead, they would write technical notes and publish them to the corporate site to cover the perceived gap in documentation.
The recommendation

• Our consulting team recommended DITA.
• We launched an evaluation process.
• It appeared to be the answer to all of our woes. We were very excited!
DITA: A super quick and not boring intro
A definition

• DITA, which stands for Darwin Information Typing Architecture, is an XML data model for authoring and publishing (Wikipedia).
• It was first developed by IBM, then handed off to the open source community.
• Today, DITA is an open standard for structured authoring and is maintained by the OASIS consortium.
A few key features of DITA

• Information typing:
  – All content has a type.
    For example, concept, task, and reference.
  – Each type has certain attributes.

• Modularity:
  – Each item you create in DITA is a component that can be re-used.

• Inheritance:
  – New components inherit the attributes of their parents.
**DITA’s raison d'être**

- **Content re-use:**
  - By far the biggest motivator to use DITA – write once, use many times.

- **Localization cost savings:**
  - Translate each component only once. (If it appears in 5 docs, that’s 4 times you don’t have to pay for it.)

- **Other ROI arguments:**
  - Eliminate formatting chores. All formatting is applied when you publish.
CCMS: single sourcing backbone

• Every component has an ID and is searchable in the "component content management system."
Also critical - single sourced output

• DITA allows you to produce multiple types of output from the same source.
Under the hood
The cost of implementation

• We learned that acquiring the free open source DITA code was just the beginning.
• The facts were daunting. We would need:
  – A DITA CCMS.
  – Supporting software tools and custom code.
  – A development team to help with our implementation.
Those costs add up

- **Rough estimates, first year:**
  - **DITA CCMS:** $30,000 - $100,000  
    (Depends on brand, features, whether installed or SAAS and amount of content, which increases over time)
  - **DITA editor:** $2,500 - $5,000  
    (Depends on # of seats, and writer vs. editor privileges)
  - **Consulting services:** $25,000 - $100,000  
    (Depends on how much hand-holding and training you need)
  - **Style sheet coding:** $15,000 - $30,000  
    (Depends on whether you need HTML + PDF)
  - **Content conversion:** OUCH  
    (We were quoted $6,300 for 400 pages. But we had thousands!)
Second year and beyond $$

- Some costs were ongoing:
  - DITA CCMS: $30,000 - $100,000
  - DITA editor: $2,500 - $5,000
  - Consulting services or staff: $20,000-$100,000

- Our determination:
  - Due to our complex doc set, a full time technical expert was the most cost effective option*.

*The math: Our consultants were charging $250/hour. For $100k, we could get 10 full weeks of consulting time or a dedicated employee for 40 hours a week for a full year.
Complexity

• DITA is simply complex
  – The implementation would take 6-12 months in the initial phase. We would then be *limping* along.
  – We were paying consultants just to get to the point of understanding what we needed to do.
  – An actual quote from a DITA blog: “*I am preparing a half-day seminar on DITA for documentation managers and I want to stay away from all the technical details - as that will definitely scare them off.*”
DITA adoption stats

- DITA has been available as a formal open source standard since 2005.
- In 2015, IXIASOFT (a DITA CCMS provider) reported that out of 150,000 technical writers on LinkedIn:
  - 8,000 said they know DITA
  - 1,200 said they are using DITA right now
- We had difficulty finding good examples of DITA implementations. The ones we saw were ugly.
The sanity check
We had to take action!
We made a pitch to management: we needed a technical solution to our highly complex content problems. It was going to cost a pretty penny.
They approved our six figure “DITA budget” to get our content under control.
And that’s when we got cold feet

- The costs, the complexity and the timeline were all daunting.
- We stalled!
We took another look at Flare

- In contrast to DITA, MadCap Flare had a very low barrier to entry:
  - Immediate download, free trial, lower cost per seat
  - We could train as a team in about three days
  - Flare had an intuitive built-in interface
  - There was no extra software required for launch
Flare offered single souring

- We had only dabbled in Flare. We were pretty clueless about its full range of capabilities.
- We discovered that single sourcing was built in, not only for conditionalizing content but for producing both PDF and HTML output.
But what about a CMS?

We were still in a quandary!

– We believed that to fully control our content, we needed a content management system. (MadCap Central wasn’t born yet!)

– How could we manage all of our hundreds of documents and set up a content re-use paradigm without one?
The turning point

• In the spring of 2015, I attended two conferences one week apart:
  – The CIDM DITA conference
  – MadWorld

• I didn’t think Flare was the answer. But I attended the conference in the off chance it could be a game changer.
Armed with knowledge

• I attended every relevant presentation, and asked questions of the speakers.
• The ah-ha moment came in the hospitality lounge where I learned more about Flare’s capabilities:
  – Inheritance can be established using a global project. (In fact, you can set up a global project like a CMS!)
  – Flare has built-in support for multiple source control systems, enabling writers to share content.
  – MadCap offers contributor/workflow software.
11th hour decision

- During the conference, I called the home office and halted our DITA CMS purchase.
- Then I returned to the team and we had a powwow. We decided we could get most of what we needed from MadCap Flare.
We rolled up our sleeves

• Within a few months we had designed a UI, hired Scott DeLoach to train and assist our team, and had everyone up and running.

• We assigned one of our team members to become our Flare guru and administrator.

• We met weekly to establish our methodologies:
  – Tagging, conditions, use of variables
  – CSS requirements, etc.
Retrospective

• Two years later, we have converted all of our high priority documentation to MadCap Flare.
• All documents use the same CSS and the same tags.
• We structure our content using concept, task and reference topics in MadCap Flare.
• The team collaborates and helps one another constantly. They are engaged.
What about a CMS?

- Our admin manages all shared items in a global project:
  - Templates, CSS, front matter
- The writers download the global project each time it revs.
- Each product line documentation set is in its own project for topic sharing.
  - Largest project contains 26 highly conditioned docs
- We bind to SVN; writers can access all content, if needed.
What we gave up

- Unique content component IDs and a searchable database.
- Automated and enforced consistency.
- Enforced structure.
  - We have chosen to follow some DITA structure, but there is no validation to ensure we do it in a specific, prescribed way.
What we gained

• Speed!
  – We got up and running and migrated all of our content very quickly with Flare.

• Flexibility!
  – Not being forced to follow certain structural constraints gives us some creative license.
  – We completely tailored our UI to our brand.

• Cost savings!
  – The money we saved on paying for professional services covered all of our conversion costs.
Thoughts going forward

• Madcap tools continue to evolve – constantly!
  – By contrast, DITA toolkit updates are every ~5 years.
• We are not yet using MadCap Lingo
  – We only translate a fraction of our content. Currently, we export to XML and reimport the translated version.
• We use Confluence more than Contributor.
  – We needed a conversational environment. But it has pros and cons. (See my other presentation.)
• We plan to fully evaluate MadCap Central
  – Can MadCap Central close the remaining gaps?
Summary

• DITA is a sophisticated open source toolkit for structured authoring.

• While DITA is free, everything about it – from consulting and special software development to DITA conversion – is complex and expensive.

• Flare offers much of the functionality at a fraction of the price, and MadCap continues to enhance features and offer new products.

• You can use DITA methods outside of DITA!
Questions?
Resources

- DITA – Know what you’re getting into: 
  http://www.writingassist.com/newsroom/dita-know-what-youre-getting-into/
- 10 Reasons for Moving Away from DITA: 
  http://idratherbewriting.com/2015/01/28/10-reasons-for-moving-away-from-dita/
- Getting Started with Topic-Based Writing: 
- Primer on concept, task, reference topics: 
  https://technicalwritingtoolbox.com/2012/05/31/difference-between-task-concept-and-reference-topics-in-dita/