acrolinx

Better Content. Faster.

A Content Life

An interview with technical content
professional Michael Iantosca
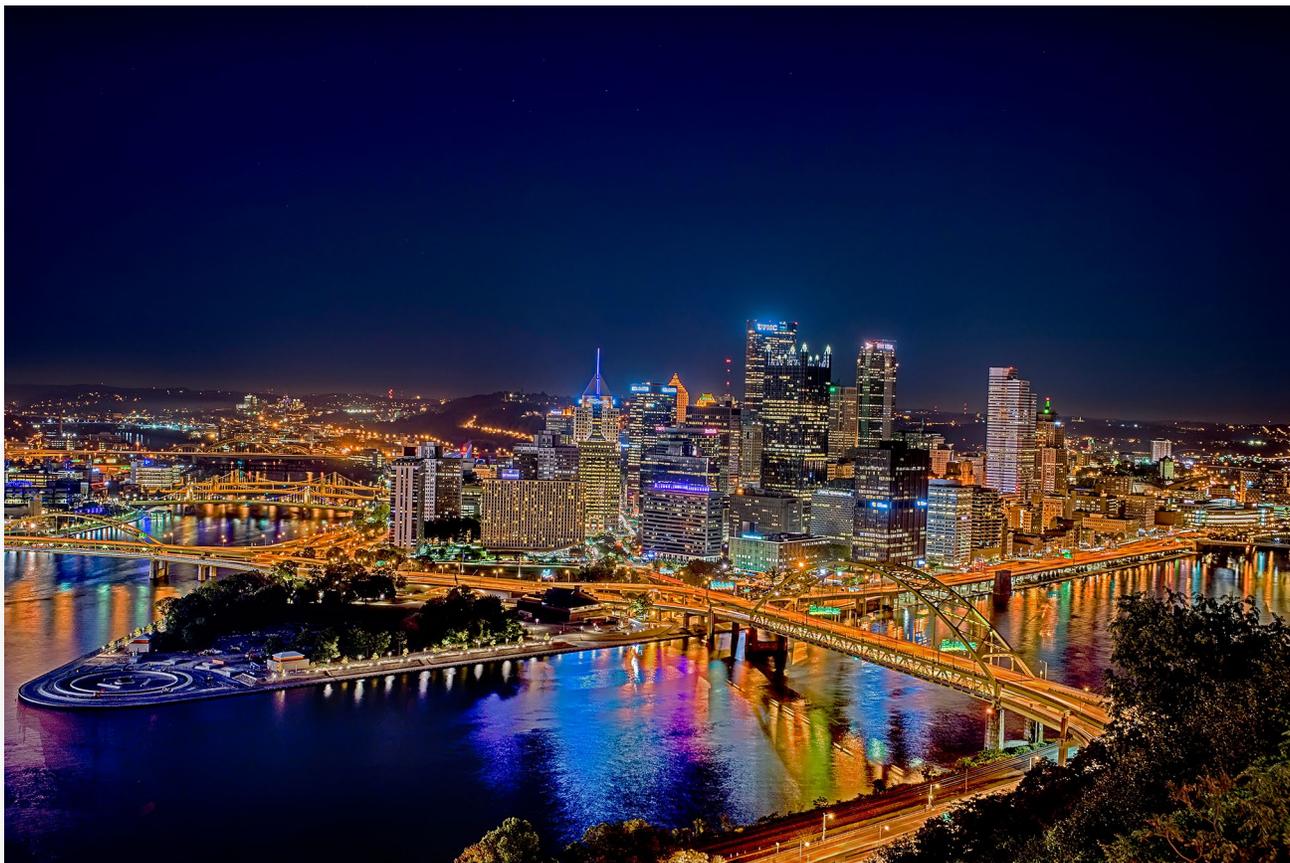
Image: © 2021 Michael J Iantosca

At age 12, Michael (Mike) Iantosca developed and printed his first roll of film. That started a lifelong interest in photography, and one might say, indirectly led him to an illustrious career in technical content. It's true (although perhaps we're slightly biased) that Mike is a bit of a "content superstar" — attributed partly to the fact that without Mike's contributions, the world of DITA content might not exist.

At the time Mike entered the workforce, he was on his way to becoming a news producer, having majored in photography at Bard College, before changing to communications and journalism at Marist. Instead, he "became intrigued" with computers while at Marist and ended up returning to Marist to work on a master's degree in Information Systems.

When Mike saw a poster advertising a position at IBM in the photography and media department, he felt compelled to apply. The position turned out to be so much more than taking photos. It was the start of a 40-year career as a content professional.

Eventually, the company began to outsource visual operations, and Mike's background in journalism, communications, and computer science made him an attractive candidate for the Information Development team (tech doc department) at IBM. Soon after, he began "writing five-pound manuals for mainframe operating systems including VM and MVS - now zOS."

After four decades of a career in the world of technology and content, you'd forgive Mike Iantosca for looking forward to retirement. As it turns out, Mike's energy and passion for solving complex content problems is stronger than ever. He recently started a new chapter as a Senior Director of Content Platforms at Avalara, helping to build a state-of-the-art intelligent content supply chain to help the leader in business tax compliance software manage its stellar growth.

For this article, we asked Mike to share his enriching insights into the world of content creation and management. We hope that his reflections, achievements, challenges, tools, and mindsets are useful to our readers, technical and otherwise!

01

Q: DITA was originally developed within the technical publications department at IBM. How did you contribute to the creation and implementation of DITA as we know it?

A: DITA almost didn't happen. I was present at the Standardized General Mark-up Language (SGML) '97 conference in Boston where the XML 1.0 specification was announced with great fanfare on stage by the likes of Jean Paoli, Jon Bosak, and Tim Bray.

When I returned from the SGML '97 conference, I immediately began to evangelize XML throughout IBM. I published an extensive white paper about XML that was widely circulated internally and I presented it to then CEO Louis Gerstner, in person. I was convinced that the next generation of our publishing platform needed to be based on XML.

At the time, the senior corporate management team had committed to build the next generation platform for web content on yet another SGML dialect called WebDoc. I convinced my immediate manager to form a workgroup to develop an XML implementation instead. The workgroup, convened in the late '98 timeframe, consisted of 10-12 folks who spent an intense year-plus developing DITA.

I consider two of my colleagues in particular, Michael Priestley and Don Day, as the thought leaders and primary inventors of DITA. I built the new platform to put DITA into production. A new corporate manager joined the team on occasion halfway through the workgroup effort and, as a result, only a handful of the workgroup members were named on the DITA patent; the rest of us received T-shirts with the new DITA logo as a thank you.

A few years later, DITA became the most successful OASIS standard in history. Had I not prevailed and formed the workgroup that created DITA, the entire industry as we know it might not exist.

02

Q: In your blog , you say that for roughly 20 years, you "led the development of multiple generations of structured content management systems and supply chain at IBM." I'm not sure all of our readers have an intuitive understanding of what a content supply chain is. Could you explain it briefly?
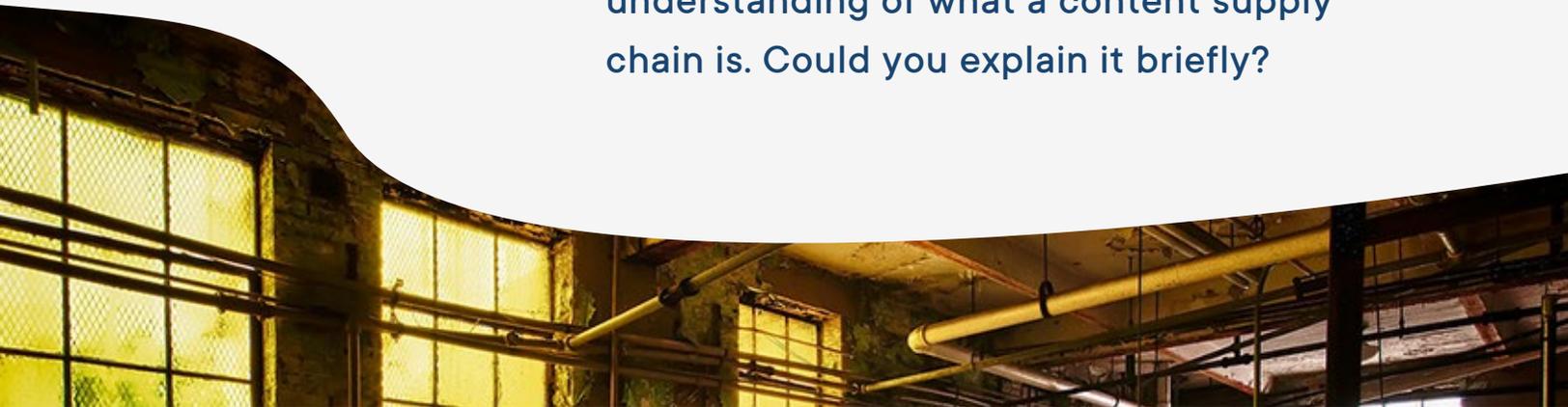
Image: © 2021 Michael J Iantosca

A: A supply chain comprises an entire ecosystem of content, standards, and technology that forms the end-to-end content lifecycle. A content supply chain treats content as a utility — where content is electricity and the technology is the grid. Some call it Content as a Service (CaaS), but it's more than that.

Developing a comprehensive content supply chain requires intense collaboration between the content operations and content engineering teams. An innovative company called Simple[A] has coined this model the Content Services Organization (CSO) that's worth reading about. In a successful CSO, everyone, including the engineering team that builds the platform, is every bit as steeped and skilled in content strategy and business goals.

03

Q:

The concept of modular content is fairly new outside of technical documentation. Marketing teams are just starting to see the benefits of modular (or componized) content creation. What took them so long?

A:

Many organizations viewed content as an expense — a necessary evil as opposed to being the competitive asset that it actually is. Content, even post-sales technical content, is a major conduit to sales and revenue. Most sales begin with subject matter experts researching prospective technology long before they contact a sales rep.

Another "AhHa!" moment happens when business leaders learn that content needs to be personalized, now that it replaces a large swath of direct selling. Big, monolithic, one-size-fits-all content can't get that done.

Don't get me wrong, content efficiency and savings still play a significant role in justifying the transition to componentized content. Content reuse, and repurposing that componentized (modular) content, enables consistency and stems the linear growth of needed writing and editorial resources. But, it's what componentized content offers (to solve the large business problems that executives tend to focus on) that seals the deal.

Image: © 2021 Michael J Iantosca

Q: What should the goals of technical/ software documentation be? In other words, what should it seek to do? What should it not do?

A: The most fundamental goal is to maximize the time-to-value proposition for customers. The way to delight customers is to help them achieve maximum value in the shortest amount of time with the least friction — revenue and profits then follow.

I see two modes of content. The historic mode is what we labeled "failure-mode" content. The technical documentation was used mostly as a fall-back when the customer had a problem. The second, forward-looking mode is proactive and assistive content. The gulf between failure mode content and proactive/assistive content has been a difficult one to cross, but many are beginning to make significant headway by using "intelligent content."

Intelligent content is modular, structured, and reusable.

It separates format from presentation, and is semantically enriched, making it highly predictable for machine processing and automation.

The industry did a wonderful job moving to task-oriented, componentized content over the years, but now needs to advance to highly personalized scenario-based content design and delivery. Achieving that requires entirely new content models and strategies that use intelligent content assets, such as taxonomies, ontologies, and knowledge graphs. These assets drive AI and machine learning to enable dynamic assembly and navigation of multiple tasks to solve complex solutions.

Image: © 2021 Michael J Iantosca

Q: How can intelligent content delivery modules be applied to product documentation?

A:

I'm most excited about moving from intelligent content to cognitive content. Countless companies have moved or are in the process of moving from monolithic and static content models to component-based, dynamic models. For that you need modular content.

Just as we migrated away from the book and chapter model to topic-oriented content models, we now need to add what has been coined "micro-documents." A micro-document model is one where the smallest usable chunk of content within topics can be encapsulated as an object.

The objects are then ready for machine-driven discovery, retrieval, assembly, and delivery to provide everything from a precise answer to a highly personalized collection to solve a complex business scenario. It's not an either-or proposition. We have the means to create topic-oriented content, and within topics encapsulate subtopic elements to which we can also apply metadata for machine mining and delivery, just as we do with whole topics and collections. Micro-documents can combine with topics and collections to assemble and deliver content solutions — even solutions that address complex multitask scenarios.

Q:

Is there a different content strategy for modular content for marketing v.s technical content?

A:

Yes and no. On one hand, technical content is different from marketing content. Both content types use different writing styles and presentations, and target different personas and cohorts. But, they share the same content and experience journey. The grave mistake most organizations make is to segregate presales and post-sales content into different delivery silos. When it comes to complex products and services, the end-to-end content journey needs to be integrated.

A typical content journey might be something along the lines of Discover, Learn, Try, Buy, Use, and Advocate — or some variant of that model. It's a model for organizing content that includes marketing content, along with content from all the other content pillars. Do customers traverse a journey model in a near, linear way? Of course not! In actual use, the path through a content journey looks more like a bowl of spaghetti. But that's not the point.

The purpose of a content journey is to do three things:

1. Ensure the main content plan doesn't have any gaps. A content journey can be used to map every content deliverable for a product, service, or campaign from content providers that span the enterprise. Some call this an Integrated Information Plan (IIP).

2. Part of building that journey involves defining all of the content types that will be later mapped against the journey, and standardizing on content type naming. If you do an inventory of content types in a typical enterprise, you'll quickly learn the same type of document is sometimes called several different things. Getting to a common definition of content types reveals gaps and duplication in the content journey.

3. Finally, we often talk about "Delivering the right content, to the right person, at the right time, and in the right experience." The content journey is key semantic intelligence that can be used to enable the "at the right time" portion of that vision, and becomes especially important when moving to AI and machine learning to enable hyper-personalized content retrieval and delivery.

07

Q: In the short term, is there an effective, transitional content strategy that gets at the idea of an intelligent supply chain without requiring a complete change of infrastructure?

A: Absolutely. Achieving the pinnacle of the intelligent content supply chain is a progressive journey in and of itself and takes time and investment. But, the size of the organization and content corpus is largely irrelevant; the effort is the same. I've done it for a massive multi-billion dollar, multinational company, and I'm now doing it again with an organization a fraction of that size.

Even if your ultimate goal is AI-driven cognitive content, you have to do the foundational work. Componentizing content is foundational, but even more fundamental is ensuring high-quality and consistent content. This is where assistive computational linguistics becomes essential. If your content is unclear, inconsistent, and confusing, no advanced search or delivery technology in the world can help recover the credibility lost by customers. Platforms like Acrolinx make that easy to accomplish and make content creators far more productive and satisfied.

Just as critical as defining content models and types, is developing a primary style guide and main terminology base. Without those, you can't progress to the next level — adding in content intelligence as metadata for machine automation. I've consulted with several companies that began a journey down the cognitive content path and stopped, just to go back to remedying these fundamental areas before moving forward — which would have had disastrous results had they not done so.

Q: What was the biggest evolution of content (or the value placed on content) that you've witnessed in your career?

A: No doubt the watershed change was the invention of structured content, which is now often referred to as intelligent content. When IBM separated content presentation (how content is formatted), from what it is (declared intent), a fuse was lit that still burns as hot and as bright as ever today. That's an example of invention-by-necessity that began with the invention of generalized mark-up language (GML) invented by Goldfarb, Mosher, and Lorie in 1969 (get it, G.M.L?).

GML was created to solve the complex and costly problem of developing a custom formatting language for every large printer IBM built and sold. It abstracted those languages to support any printer by declaring intent instead of one-off machine-specific languages for every new printer that were large and each cost a small fortune.

GML set the stage for its more highly structured variant called Standardized General Mark-up Language (SGML), invented by Dr. Charles Goldfarb. SGML became an ISO standard in 1986 — which is when I entered the fray when Dr. Goldfarb unexpectedly dropped by my office in the IBM programming lab. He wanted to review a patent submission I had done using GML to launch embedded media players with the then-emerging electronic book technology. That was at the genesis of the micro-computer-based multimedia age. GML and SGML preceded HTML and the web by many years, and they set the stage for implementation of practical hypertext.

You could confidently say that Goldfarb, Mosher, and Lorie were the grandparents of the web as we know it today, and Dr. Goldfarb the grandfather of XML that followed. Finally, the evolution of DITA built upon all of that to create what we call Intelligent Content today.

Image: © 2021 Michael J Iantosca

09

Q: When did you first become aware of Acrolinx and what was it about Acrolinx that stood out to you?

A: I first engaged with computational linguistics in the mid-90s, working with IBM's top linguistic scientists at the famed IBM Watson Research center in Yorktown Heights, NY. Some of the research scientists there were every bit the stereotypically brilliant mad-scientist types in the movies. The team had created a service called Easy English Analyser (EEA) that analyzed content and provided recommended corrections and improvements for grammar and style — not dissimilar to what Acrolinx does at its core. It was extremely powerful, but shall I say, austere. At the Watson Research Center, we spent a day defining APIs so that we could use the service with our content authoring tools, and it worked. But, it wasn't user-friendly and had limited ability to customize. Like much research at IBM Watson, EEA wasn't intended to be productized as IBM wasn't in that business. An interesting note, however, is that some of that team's same research made it into what became the core of IBM Watson, IBM's highly successful AI platform.

When I learned about Acrolinx in the mid-2000s, I immediately jumped to learn more. As the IBM technical content tools lead, I was under extreme pressure from the IBM editorial leads — some 125 of them at the time, for some assistive technology for editors. Up to that point, we provided very little tooling for human editors. Also around the same time, there was an Acrolinx sales lead speaking to individual content teams. After seeing the depth and breadth of what Acrolinx had to offer, as well as its future potential, I asked the Acrolinx sales lead to cease and desist reaching out to individual teams and coordinated a comprehensive pilot study that culminated in the purchase of 3000 named user licenses.

We deployed Acrolinx to virtually every organization in the enterprise that produced customer-facing content and also to several internal-facing content teams. The product content alone consisted of tens of millions of multilingual pages, published in dozens of national languages, in support of thousands of products, with millions of visits per week.

The use of Acrolinx began with product technical documentation, then expanded to include marketing, sales, developer support, partner support, learning, technical support, and many more. At its height, we had 5000 content creators using Acrolinx, 15 harmonized and well-governed style guides, and an enterprise terminology corpus consisting of 1.5 million multilingual terms. Using Acrolinx, we were able to drive measurable content quality up from scores in the low 60s to the high 80s on average — and that was mostly on content previously edited by professional editors.

Q:

To any technical writers or content writers at the beginning of their career, what advice would you give them to be better writers?

A:

You might be surprised that I don't immediately dive straight for some advanced technology, strategy, or methodology, but at my core I'll always be a writer first and foremost.

There are three basic axioms that are as old as the hills of which we must never lose sight, even as we lurch towards the age of cognitive content, AI, and machine learning — even for new technical writers.

▶ *Know your audience.*

▶ *You can't not communicate.*

▶ *Recognize and immediately document your perplexities.*

The first of these, "know your audience," is as fundamental for writers as breathing is to humans. It will never go out of style, as our core mission is to assist humans, and even machines when we get to autonomic content, to accomplish their objective and goals as quickly and with the least amount of friction. You can't do that without knowing your audience and their goals.

The second of these "you can't not communicate," is one of the five axioms of communication defined in 1921 by Paul Watzlawick, based on the truth that humans communicate as soon as they perceive each other. This also extends to content and machines. Even when we get to cognitive content, we need to remember that every user that comes to your website or application has already provided a flood of utterances. And the more they interact (or fail to interact), the more we know about them from a personalization perspective.

As we forge ahead into the era of cognitive content, these three points remain as fundamental and essential as ever. Beyond these are the best practices of

minimalism, progressive disclosure, and task-oriented writing, which are all staples of the profession.

Finally, as content professionals, we're constantly both students and teachers. In our zeal to assimilate complex subject matter to communicate with others, we must remain ever-vigilant to not dismiss what we find initially confusing.

Moving ahead, we have to progress beyond intelligent content. Many have only gone half the distance in the content marathon. The next frontier requires knowledge of intelligent content assets that complement intelligent content. Intelligent knowledge assets include taxonomies, ontologies, knowledge graphs, computational linguistics, AI, and machine learning to name a few. These are the building blocks of the next generation of content and user assistance — and many are already behind. Great tools help create great content, and the use of platforms like Acrolinx are going to play a central and essential role in building that future.

acrolinx

Better Content. Faster.

www.acrolinx.com