

Making XML Work

*Seven reasons why it doesn't and
ten ways to make sure it does*

Bill Kasdorf

Vice President, Apex Content Solutions

General Editor, *The Columbia Guide to Digital Publishing*

**Two recent
direct quotes
from university
presses:**

**“I know we should
have XML . . .
I just can’t figure out
what for.”**

**“We asked around . . .
other presses aren’t
having much success
getting XML from
their composers.”**

The Main Reasons for Failure



1 Model is too complex for comp's toolset

- Need to adapt the tagging requirements to what the *tools* can handle and what the *people* can handle.

Don't build a cathedral and hope everybody will sit in the pews and worship. . . .

Commercial publishing is full of examples of this.

The Main Reasons for Failure



- 1 Model is too complex for comp's toolset
- 2 Lack of collaboration
 - “Here’s my DTD. Give me XML!”

What the comp says:

“Sure, we can do that! Whatever you want!”

What the comp thinks:

“Please don’t look behind the curtain. . . .”

The Main Reasons for Failure



- 1 Model is too complex for comp's toolset
- 2 Lack of collaboration
- 3 Lack of documentation
 - A DTD is not enough; you need use cases, examples, style guidelines

“Oh, you mean your heads are typed in Title Case, and you don't actually tag all the names and dates?”

The Main Reasons for Failure



- 1** Model is too complex for comp's toolset
- 2** Lack of collaboration
- 3** Lack of documentation
- 4** Haven't optimized the workflow
 - What's best done as pre-processing?
 - What can editors best do?
 - How can the tagging fit the comp tools?
 - What's best done as post-processing?

The Main Reasons for Failure



- 1** Model is too complex for comp's toolset
- 2** Lack of collaboration
- 3** Lack of documentation
- 4** Haven't optimized the workflow
- 5** Thinking that parsing = quality control
 - Lots can be wrong in "valid XML"!
 - Need to check for style issues, completeness

You don't want to find the problems on your website!

The Main Reasons for Failure



- 1** Model is too complex for comp's toolset
 - 2** Lack of collaboration
 - 3** Lack of documentation
 - 4** Haven't optimized the workflow
 - 5** Thinking that parsing = quality control
 - 6** Chose model arbitrarily
 - Models are created for specific purposes
- "But everybody else seems to be using that DTD."*

The Main Reasons for Failure



- 1** Model is too complex for comp's toolset
- 2** Lack of collaboration
- 3** Lack of documentation
- 4** Haven't optimized the workflow
- 5** Thinking that parsing = quality control
- 6** Chose model arbitrarily
- 7** Model constantly changes
 - Need disciplined, documented version control

**It doesn't have to
be that way!**

**The Top Ten
Questions to Ask
to Make Your XML
a Success:**

1. What do you want XML for?

- Rendering on a screen?
- Archiving? Future proofing?
- Generating new revenue? Slicing & dicing?
- Subscription based products?
- Licensing? Aggregators? Resellers?
- Attracting customers? Authors?
- Collaboration, scholarship?

2. How do you **organize** your content?

- Books vs. journals
- Monographs vs. reference works
- Discipline-specific groupings
- What belongs together?
- What do you distribute/sell to the same people?

It may be better to have more than one model than to shoehorn everything into one generic model.

3. How do your **users** relate to your content?

- How will they find it?
- What questions will they ask of it?
- What do they call things?
- What will they search for?
- What “chunks” will they want?
- What models do they expect?

Make your content magnetic, clairvoyant, & sticky.

4. What do your business partners need?

- Will they need to transform what you give them?
- What does that cost?
- How should the components be named, organized, packaged, and documented?
- What metadata would be helpful?

Don't be seduced when they say

“we'll just take it however you've got it.”

5. What can your vendors do for you?

- Most leading comp firms are expert at XML
- They provide content to many outlets
- They deal with many models from many pubs
- They've seen most of the pitfalls and gotchas

*Beware of the “vendor mentality”:
they'll do what you ask even if they think it's stupid.*

6. What **standard models** are appropriate?

Models are created for *purposes* and *communities*.

- **NLM**: Now de facto model for **journals**
- **DocBook**: optimized for **technical documentation**
- **TEI**: dominant in **humanities scholarship**
- **EPUB**: **Delivery** model for **e-books**
- **DTBook**: The standard for **accessibility**

7. How can these be adapted to your needs?

Models need to be adapted to:

- Fit your **publications** (e.g., add/remove elements, change names, make more restrictive/permissive)
- Fit the **capabilities** of your tools, staff, vendors
- Optimize various stages of your **workflow**
- Enable creation of **products** that delight **customers**

8. What can your **tools** and **technologies** do?

- **Styles-based systems** (Word, InDesign) need a flatter, simpler model; can't generate/rearrange text, handle deep hierarchies, format on attributes, etc.
- **High-end systems** (3B2, XPP) can do those things
- **Things get better:** NLM template for Word 2007; InDesign outputs EPUB; Word has "Save as DAISY" option; all Office 2007 is XML as basic file format

9. What can be **automated** and what shouldn't be?

- It's almost never all one or the other
- **Computers help people**: optimize what's done by computer and what's done by human beings
- **Customize** the tools & processes for the people
- **Monitor** how it's working: find repetitive tasks to automate, reveal hidden problems or issues (e.g., use colors to show fig/table references in proofs)

10. What can each party in the workflow do best?

- **Don't expect much of authors.** Just get their content; use your staff and vendors to get things right.
- **Copyeditors** don't have to **clean up/prepare/style files**; that's better done by file prep dept. or vendor.
- Copyeditors are good at **classifying, linking, etc.**
- Don't forget about **art prep**: integrate w/ edit prep
- Some things are best done **post-comp.**

Thanks!



Bill Kasdorf

Vice President, Apex Content Solutions

bkasdorf@apexcovantage.com

+1 734 904 6252