

Trends in XML Content Management (for Publishers)

Lisa Bos

Really Strategies

lbos@reallysi.com

December 4, 2007

Introduction



CMS Consultants

REALLY [] STRATEGIES™



CMS Vendor

[] RSuite
CMS

Outline

- In 2000, publishers knew they needed to do something online, but they didn't know what.
 - They thought XML might help.
 - XML did, XML technology often didn't.
- In 2007, publishers have lots of specific product plans.
 - They already do XML.
 - The technology helps a lot and is getting better all the time.

My premise

- The cycle of the publishing industry is also the cycle of the individual publisher

Don't know much



Make mistakes



Learn through pain



Become productive



Meanwhile: Technology gets better (a lot better)

Overall 2000 context

- A few publishers had done SGML right and were successful
 - Reference content
 - Automated print
 - Automated digital (CD/Web)
 - Print still drove update cycles
- Everyone else was trying to figure out what to do about electronic publishing

Request: Forgive any gross generalities about publishers.

Publishers – 2000

- Publishers were:
 - Hopeful
 - Skeptical
 - Naïve
 - Inexperienced (with XML)
 - Unsure about business models
 - Busy
- We had to convince people that XML was worth it

Publishers – 2007

- Publishers are smart and experienced (everyone)
- Some IT departments still think XML is just data with tags
- Nobody asks for ROI justification for XML – just where it should come in the workflow
- Business models in motion, but customers know where the money is
- Pragmatic
- Really busy

Processes – 2000

- Print was first
- Web was scary
- Only a brave few were in the Web group, some were still stuck in the CD-Rom group
- “That guy doesn’t use a computer yet”

Processes – 2007

- Print is mostly first (usually because it's harder)
- But many publishers want both/either-first
- Editorial is starting to care about online content and customers
- XML moving up in the workflow

Stated requirements – 2000

- “We need to go to XML” (or maybe SGML?)
- Magic conversion to XML
- We’re the web group, and we want a system for our print group
- Media-neutral environment, but ...
- Can’t we just use Word?
- We want one big system for everything.

Stated requirements – 2007

- We need XML earlier in our workflow
- Can't we just use Word?
- We want to start small and be able to grow/change
- Our users are all over the world
- Our customers are CMS users too
- We don't want to choose whether we're going print-first or web-first. And that doesn't mean we want media-neutral.

Implied XML requirements – 2000

- DTDs
- Metadata inside the XML
- Element level management
- Ongoing configuration
- Friendly XML editing in forms and in documents
- Slicing and dicing

Implied XML requirements – 2007

- Lots of schema languages
- Ongoing schema changes
- Querying XML
- Metadata inside XML, XML metadata for non-XML
- Element level management (versioning, re-use, linking)
- Ongoing configuration
- Friendly XML editing in forms and in documents
- Rapid XML transformation and mashing up for delivery
- And then there's DITA

CMS technology – 2000

- Web web web
- Print production systems (QPS)
- Document management with bolt-on XML
- A few SGML/XML systems, some with performance issues and without features needed by publishers
- Two editors (Epic/Arbortext, XMetaL)

CMS technology – 2007

- Web CMS heading towards XML CMS
- Formerly niche XML/publishing players coming out of their niche
- New generation of native XML CMS (like RSuite)
- A bunch of editing options
- Related print capabilities: XSL-FO engines, XML in/out features in desktop (e.g., InDesign) and batch composition (e.g., XPP) tools

Actually Implemented – 2000

- Tried to get customers to focus on process and building internal expertise, but often ended up with:
 - Long processes for CMS selection/design
 - Over-engineered DTDs (IMO)
 - Mostly custom systems or Documentum
 - Big, expensive, long projects that died under their own weight (for business reasons)
 - Editorial tools that people hated
 - Reasonably effective print automation for low-design publications

What really worked best – 2000

- Targeted experiments with expert users
- File system instead of database
- Vendors doing the work
- And, occasionally, a larger custom system because the customer was really smart
- But some of it was ugly:
 - Metadata in relational fields when it shouldn't have been

Actually Implementing – 2007

- Multiple integrated systems
 - Print production systems (like K4)
 - Native XML CMS (like RSuite, Docato)
 - MarkLogic + Documentum
 - Documentum
 - Digital asset management systems (DAMS like Artesia)
 - Custom systems
 - Web CMS (especially for user-generated content)

What really works best – 2007

1. Native XML for storage

- Query your XML
- Combine XML queries with fulltext search
- Leave your metadata where it is
- Reconfigure your system (including schemas) as needed
- Enrich content on an ongoing basis

What really works best – 2007

2. A variety of editing tools/approaches tailored to the content, user, and workflow
 - XForms
 - Wider variety of fulltext editors: XMetaL, Arbortext, xopus, oXygen, Serna
 - Adobe Creative Suite
 - Web widgets

Both/either-first approach

What really works best – 2007

3. Tighter integration with rendering tools

- InDesign/InDesign Server, Typefi
- XSL-FO
- XSLT
- XQuery
- Web CMS

What really works best – 2007

4. Each solution in its place

- Print production
- Individual CMS's
- End repository
- Web CMS

(Although in this case, “best” is really just “better”)

What really works best – 2007

- Bottom line: Technology isn't usually the barrier to product/content development before anymore.

What else is coming

- Real link management
- Give your internal users the same kinds of web-based tools you gave your customers
 - UI – Navigation – Community
- Real collapse of the external/internal divide
- Richer editor-centric experience
- Powerful CMS tools for DITA (mostly been in the authoring tool so far)
- Integrated OOXML (Office 2007) solutions
- Better auto-classification/taxonomy tools

What's happening in the vendor space

- It's always been messy, it will get messier
 - Print production systems gaining in web features
 - Web systems gaining in “enterprise” and XML features
 - XML systems getting deeper XML and gaining web features
 - Enterprise systems adding native XML (Documentum + X-Hive)
 - Tech publications vendors adding DITA and some XML
 - Lots more hosted niche and not-so-niche solutions

Questions?

Lisa Bos
lbos@reallysi.com