

The Brief, Tortured Life of XMP

BY RON ROSZKIEWICZ

Like any concerned relative, I've been watching the adolescence of Adobe's Extensible Metadata Platform (XMP) with a mixture of hope and disappointment. XMP showed such promise early on, but like many a precocious child, XMP has grown into an under-achieving teenager (in technology years of course). It's sad to see such potential go to waste.

All of us who have followed the growth and adoption of XMP technology from a Seybold Hot Pick in 2001 to the present have no doubt wondered when the "tipping point" would occur — if at all. When will the technology become so compelling that the spark will burst into wildfire and we'd be off and running on another PostScript/PDF adventure? My guess is that Adobe is watching the same developments and is also wondering when the wheels will stop spinning and gain some traction.

The truth is, Adobe could be doing a lot more to create the conditions that converge in a tipping point. The entrepreneurial, market-aware side of the company is at odds with the institutional, myopic side, and result is a stalemate.

At this point, you might be wondering, What is he talking about? More and more vendors are announcing support for XMP in their applications and Creative Suite 2 has new metadata functionality sprinkled throughout. Although it seems that the tipping point should be just around the corner, after years of waiting, I doubt it.

Adobe offers two forms of software developer's kit (SDK) for anyone to use without any license fees. It also uses the technology in its own applications. The marketplace has recognized that metadata is the workflow lubricant that will be used by more and more creative managers, IT professionals and system developers. The XMP platform itself is built on industry standards, such as XML and RDF, and they seem to be thriving. So what is preventing this technology from rocking, much less tipping?

There are a number of issues, beginning with a lack of support and discipline

during XMP's crucial formative years. XMP represents a brilliant conjunction of technology dots, but from the outset has received only tepid support from Adobe.

Although I have no firsthand knowledge about what went on in early engineering and quality assurance meetings, I can speculate based in hindsight. XMP, as implemented in the first round of applications, was a limited and easily controlled experiment. Bits and pieces of different industry standard schema were tacked together into a set of common denominator schema properties. Some of the stuff even got used. But at the same time, Adobe released a developer's kit that made possible the creation of custom panels, custom metadata fields with unique values and any several ways to extend the initial experiment.

The problem was that no one bothered to test whether any of this custom stuff worked in the applications that had to support it. So Photoshop, a prime consumer of metadata, could not embed custom metadata created according to Adobe's rules. Add to this some crucial omissions from the supporting developer documentation and the result was two years of virtual stasis.

In 2003, the custom metadata problem was solved with the release of Creative Suite 1, where things more or less worked as advertised. Two years of lost innovation because of flawed technology is not insignificant, however.

Now that another two years have passed, developers are providing tepid support by reading some XMP metadata. Hardly any developers are writing XMP metadata, and no one, to my knowledge, has adopted it as the infrastructure of their system. The extremely slow pace of innovation based on XMP indicates a critical flaw in this most logical approach to managing the relatively simple concept of metadata. What's standing in the way of joint press releases from Microsoft and Adobe announcing its adoption in their applications, or Google and Adobe from heralding a new age of Internet search based on XMP?

Adobe never released a library for reading and writing XMP to files as part of

its developer's kit. As you might have noticed, the digital asset management solutions you use read IPTC, EXIF and Dublin Core, but they don't allow you to write to or edit the metadata in the file. Sounds crazy, doesn't it? Imagine "ingesting" (the meta-world way of saying importing) images into a database, absorbing the metadata properties and values into the record for that image, but not being able to edit and re-embed the edited data back into the file. Ouch! What then is the point of using the technology?

A case can be made for limited implementations, a copyright, usage, keyword case that is acceptable for most basic needs. But the inability to write XMP means that the proposed standard is not extensible. By and large, developers aren't wrapping innovative applications around the core technology Adobe supplies. Some are, but the notion of separate R&D groups re-inventing the XMP wheel in labs around the globe because of the "compelling" nature of XMP would be unproductive and signals a failure in Adobe's support for XMP.

Why hasn't Adobe already developed and released the missing XMP read/write library? One rumor has it that the problem is licensing and support: if and what to charge for the license and how much support will be required. Of course, the SDKs for XMP are free, a scenario that has played out before with PDF. The cost to license a PDF library depended on whom you talked to. Over the years, responses have ranged from "not available" to \$100,000 per license. Certainly there would be a cost to allocate manpower to create this technology that Adobe in fairness should be expected to recover — sort of.

The word is that all the technology needed was created to support this exact functionality in Bridge. Perhaps the issue is a competitive one. Bridge, developed in-house, gets first crack at full-bore XMP while everyone else gets a 2-year-old incomplete developer's. I might say fair enough if Bridge recognized custom metadata embedded using Adobe's SDK-approved File Info panels in Photoshop, but it doesn't. Adobe didn't make the leap from Photoshop embedding to Bridge recognizing and displaying. Was it tested? Probably not. (Don't bother to look through the Bridge User's Guide for XMP for guidance, there isn't one.)

So after four years, where are we? The first two years are the forgotten ones. XMP didn't work as advertised and developers disregarded it because they could not incorporate it into their solutions. Over the past two years Adobe has promoted metadata and XMP and has made visible use of it in Adobe Creative Suite applications, amid much public discussion. Lack of a read/write library in the developer's kit means that whatever adoption ensues will be limited and will be more expensive and difficult than it would have been with

Adobe's help.

In my opinion, the situation two years from now will be incrementally better than it is today. Thanks to Adobe's continuing tepid support, a promising technology will not be pushing the automated publishing envelope and will still be carrying water as a make-do digital rights and keyword container. Customers who are now beginning to realize the potential for metadata — and by extension, XMP — will continue to be frustrated with how little they can do with metadata and how much manual interven-

tion is necessary to do it. Bridge will have one more round of bug fixes, performance boosts and improvements before it returns as a turbo-charged plug-in to Photoshop. And instead of XMP, we will be using a new standard written by a Ph.D hired this week by Google. **TSR**

About the Author

Ron Roszkiewicz, is a consultant, writer and impatient software user and developer. He lives in Leucadia, California and can be reached at roszkewicz@gmail.com.