

## Ad Hoc Rules, Process Failures and OpenXML

The last minute arguments against OpenXML appear to be shaping up to be: a lack of full consideration of all comments at the BRM, no final text for approval, and errors remain in OpenXML. What has puzzled me is the response by some that attempt to dispute the factual basis for these charges.

I say that because I think it is beyond dispute that not all comments were adequately considered at the BRM, there is no final text for approval and I have no personal doubt that errors remain in OpenXML. **However**, responsibility for all of those unhappy facts do not lie with OpenXML.

Let's start with the lack of adequate consideration of all comments at the BRM. I have searched the JTC 1 Directives and can report there is **no** five-day limit on a ballot resolution meetings anywhere in those Directives. Moreover, the norm is to have multiple BRM meetings until all comments have been fully considered. The responsibility for the five day ballot resolution meeting process failure lies with someone other than OpenXML.

There is no final text for review prior to voting. True, and it is also true that the ad hoc rules in place were not written by OpenXML. Moreover, even if a final text were available, there would not be adequate time to review it. It seems churlish at the very least to condemn a proposal for the failure of ad hoc rules developed by others.

Finally, the old saw that errors remain in OpenXML. Well, I was reading the second edition of the HyTime standard recently and noticed an error in the text. That text was approved in 1997. So yes, there still errors in OpenXML. The question is: Are they significant enough to prevent it from being an ISO standard? That is the question where the anti-OpenXML forces fail and why they worry about vague and unspecified errors.

I don't think that OpenXML should be penalized for ad hoc rules/process failures that are entirely the responsibility of others. Whatever the current shortcomings of OpenXML, SC 34 is the proper place to repair it.

Covington, 26 March 2008

Patrick Durusau