EDITORIAL:

THE BALKANIZATION OF STANDARD SETTING

Andrew Updegrove

A recurring question that underlies the legitimacy of modern standard setting methods is this: "What is a standard?"

Almost immediately the definitional process becomes complex, and even contentious. Is the Microsoft Windows operating system a standard, or not? In other words, should we judge a specification only by the process that created it, regardless of its uptake (or the lack thereof), or is success the more important criterion? Most people resolve this conundrum by adding a results oriented modifier, calling widely-adopted commonalities like Windows a "de facto standard".

But what about process-based "standards"? At the traditionalist end of the spectrum, there are those that advocate the use of the word "standard" only in relation to the work product of an accredited standards development organization, relegating the work product of even the most well-respected consortia to the second-class status of mere "specifications." At the other end are those that find consortium-produced specifications, and even open source project derived work product, to be perfectly entitled to be called standards.

In our view, the essential element of a standard (*sans* the words "de facto") is that it has been produced through an open process. Admittedly, his simply shifts the discussion to what the attributes of an "open process" are. But this shift is the most vital step along a line that begins with proprietary control and ends with standards that can be safely implemented without fear that some individual company, or group of companies, can exercise undue control over the future development and permitted uses of the standard.

Of course, all this becomes only so much dancing on the head of an etymological pin if the market ceases to care about process at all. Increasingly, some companies seem to be promoting this concept.

In this issue of the *CSB*, we review the burgeoning number of situations where a company, or group of companies, creates a specification outside an SSO. In some cases, the resulting work is shopped around to existing SSOs and offered for adoption and maintenance. This may be a useful and productive exercise, where an SSO might not have wished to allocate resources to the internal development of the specification in question, but might be quite happy to accept it for ongoing maintenance. The practice can be suspect, however, if the contributors of the specification have undue influence over the SSO that accepts the grant. And even when such is not the case, something may be lost where too few points of view are brought to bear in creating a specification during its initial conception.

Recently, SSOs are sometimes being bypassed entirely, with the developers of a new specification encouraging direct adoption by the industry without any assurances of continuity or other protections at all.

Another disturbing trend is the development of specifications in too-casual and chaotic a setting, even if that setting is arguably "open" in some respects, as seems to be occurring in the case of RSS standards.

While there may be no proprietary interests at work, there are risks that specifications that are conceived in such a haphazard fashion may later prove to infringe third-party patent claims, to the chagrin of all those that have already implemented. The thousands of open source projects that are ongoing today run similar risks.

While we have always supported experimentation and evolution in the standard setting process, we believe that this trend of casualness is one that is unhealthy for the industry. Without process there is no protection from proprietary self-interest, or even from abandonment or tactical withdrawal of a specification by its owner. And while some material being offered to the marketplace may be purely functional rather than fundamental, it is easy to become complacent in accepting such offerings and incorporating them into products, perhaps failing to see the importance that these Trojan horses may later play.

Testing the envelope is how productive evolution takes place, in the virtual as well as the natural world. But not every mutation is healthy, or deserves to propagate. Those who build to a "standard" would be well advised to be sure that the process that created that standard met minimum procedural requirements.   And the industry at large would do well to discourage, rather than encourage, companies to spend their time developing specifications outside of an open process.

Comments? updegrove@consortiuminfo.org

Copyright 2004 Andrew Updegrove