
Subject: Re: Linux release cookbook

Posted by [cbpporter](#) on Wed, 30 Jan 2008 18:03:03 GMT

[View Forum Message](#) <> [Reply to Message](#)

Quote:Well, I had big hopes for LSB.... 4 years ago.

Quote:The LSB exists, theoretically it works, but nobody wants it.

Me too! I was really fed up with a lot of binary incompatibility, endless recompiles from sources and back then I was quite the Linux enthusiast (I still like it today, and use it from time to time). The idea is good, the implementation is not that great and the adoption is horrible. Even if most of today's distros are LSB compliant, that only means that other LSB compliant software would find a working LSB runtime and would have no problem running. For a software to be LSB compliant, it must use the LSB core (which is very poorly placed in the filesystem IMO, reducing the flexibility of "system" paths) and all the libraries must either be LSB compliant or static linking must be used. The last time I checked, the number of LSB compliant applications and libraries was close to zero (I mean a relative closeness: with thousands of libraries if you have 20 done, you're close to zero), and static linking is practically impossible with any non trivial applications. Try static linking with dozens of libs and you'll get huge exe sizes and a great redundancy in the binary files.

But for U++, it would be easier to produce LSB compliant programs. Reliance on external libraries is very low (I believe only some optional gtk libs for skinning, which I don't really know if there are any LSB versions of) and the rest of system calls could be routed toward the LSB runtime by using their tools and some not too hard tweaking of the build process (please correct me if I'm wrong).

While this is possible, I think the question is: who cares? Most distros don't even have LSB selected by default on their general installation option.
