

Choosing between Mallard and Ducktype

Sébastien WILMET

May 2022

Recently I wanted to rewrite a software manual to use a friendlier format. I already knew Mallard¹ (XML) that is used to document GNOME applications. But the Mallard project has also developed a new format called Ducktype², which is a *compact* or *lightweight* syntax for Mallard, similar to Markdown, reStructuredText or AsciiDoc.

Since Ducktype is more recent and a lightweight syntax seems appealing, I looked at Ducktype. My conclusion is to keep using the Mallard (XML) format for my needs at least. The reason is not because Ducktype is badly designed or awkward to use, quite the contrary, it looks a great document format.

The advantage of XML is that it's a well-known *syntax*. Between different XML formats, it is just the element names and attributes that change, but there is good documentation on the Mallard website. XML in general is a familiar way to write documents or data, because HTML has the same constructs (and XHTML is an XML format).

The problem with lightweight formats, like Ducktype, is that they all differ in subtle ways (for their syntax). Lots of people (including me) already know Markdown, because GitLab and GitHub can render them nicely when browsing a Git repository on the web.

So instead of juggling between different lightweight/compact text formats, I prefer to stay with a well-known syntax.

¹Mallard XML format: <http://projectmallard.org/>

²Ducktype format: <http://projectmallard.org/about/learn/ducktype.html>