

# TikaProposalDiscussion

Use this page or the Incubator general mailing list for discussing the [TikaProposal](#).

I am not sure if there are already some existing apache projects dealing with this licensing issue, but how do you see option where Tika would be using a build system like maven2 and would release through maven repository -> no offending code/libraries in repository nor in releases. Of the downside (atm) of this is that not all of those libs are available from common m2 repos. - Sami Siren

AFAIU we can't even **import** any (L)GPL classes in Apache code. See the draft [Third-Party Licensing Policy](#) page for the details. One way to work around this problem (and IMHO a good solution generally) would be to provide some SPI interface in Tika and use a service provider mechanism to dynamically bind to all the implementations available at runtime. This would invert the compile-time dependencies and a user would only need to add the parser libraries whose functionality is needed as extra dependencies in addition to Tika. - Jukka Zitting

AFAIU the importing is the only remaining "issue" currently (in addition to politics), if the proposed changes "go through" it is not anymore. By SPI you probably don't mean javas standard SPI mechanism, because that would need those libs to implement Tikas not yet existing interfaces? - Sami Siren

Yeah, I'm also hoping for a reasonable solution to the LGPL issue, that's one of the reasons for listing also LGPL libraries above and leaving the the actual dependency policy open to be decided during incubation. Re: SPI; yes, that was my intention. Of course we can't expect to get such phantom interfaces implemented yet, but we could well start with small separately released integration components and advocate getting them included in the upstream libraries once Tika starts gaining more recognition. - Jukka Zitting

I don't like listing LGPL'd libraries as dependencies. It raises a red flag and thus invites lengthy, non-productive discussions. We will adhere to Apache's Third-Party Licensing Policy, which means we cannot require LGPL'd stuff. A "dependency" sounds like a requirement. So, while we'll probably permit the use of some optional LGPL'd libraries, let's not list them as dependencies, ok? - [DougCutting](#)

OK, removed the LGPL libraries. - Jukka Zitting