

## Packages - Packaging Request #3493

### [flashrom-stable]: replacement for 'flashrom'

2023-06-09 03:54 AM - bill-auger

<b>Status:</b>	in progress	<b>% Done:</b>	0%
<b>Priority:</b>	bug		
<b>Assignee:</b>	bill-auger		
<b>Category:</b>			
<b>Description</b>			
from the mailing list:			
wael wrote:			
<p>After the split in the flashrom upstream, one of the old devs (Nico Huber) took to maintaining the fork which carries on with the original mission.</p> <p>This fork is called flashrom-stable and it is part of the Coreboot tree too. So far it has slightly better chip support than the former upstream and a major feature is the focus on the core mission instead of getting lost in the rust hype as happened to the upstream thereof.</p> <p>I have adapted a package build from the AUR, with minimal changes to make it work under Parabola's requirements and conditions. Tested currently fully on a ThinkPad X200 (x86_64),</p>			
bill-auger wrote:			
<p>i am mostly wondering, why should this conflict with 'extra/flashrom' - because of file conflicts, sure; but are both needed? - is 'extra/flashrom' still desirable? or undesirable, or redundant? - is rustiness it's only problem?</p>			
wael wrote:			
<p>As to why it should conflict with "vanilla" flashrom, because it is a fork thereof and they share a lot of the core code and files - that I think that is pretty obvious.</p> <p>Rust was just one example of one of the issues and why this fork exists, as it is happening now, most of the new development and work towards board support is being done on flashrom-stable (don't let the name mislead you, it is a much more active project), and in general since it is under the wings of Coreboot it has more eyes on the code and much better support - especially with Coreboot developers that are reverse-engineering a lot of boards and chips, that knowledge gets contributed in directly.</p>			
bill-auger wrote:			
<p>yes, i admitted that is obvious - but are both needed? - if they are duplicates, or if the new one supersedes the existing one, we should remove the one in [extra] - otherwise, we (you) should write a new wiki article, explaining why there are two packages of the same software, and why someone may prefer to use one or the other - so, i am asking those questions now - by conflicting with the extra package, it is suggesting that both packages should be available - so why keep two? - why are you not suggesting to remove the one in [extra] instead?</p>			
wael wrote:			
<p>As far as I see it, it is preferable to replace flashrom with flashrom-stable,</p>			

and not because of just rust (although that would mean being simpler and lighter to build).

Mostly because of being part of Coreboot and getting more development and support (as is already the case, flashrom-stable supports more target flashers and chips already), and also because the main developer from flashrom is actually now the one behind flashrom-stable (Nico Huber).

bill-auger wrote:

if it is objectively "better", then there is no reason to keep the one in extra - but in that case, that suggests firstly, before doing *any* of this, to contact the arch packager of the 'flashrom' package, and suggest to change the upstream source to the objectively "better" one - if accepted, then neither of us would need to do anything more - but perhaps the arch packager would have a valid argument against doing so - i am trying to determine now, what that objection might be (ie: what reason is there for parabola to keep both packages?)

wael wrote:

I haven't checked with the Arch packager, but I don't think they'll be replacing it any time soon - as that version is still technically the "official" flashrom.  
Should I still try to ask anyhow?

bill-auger wrote:

sure, i would try - if you could convince arch to adopt it, we could drop this discussion altogether, and have one less package to maintain - i was nearly successful doing the same for 'cowsay' last month

wael wrote:

I'll check with the flashrom packager upstream and report back.

## History

### #1 - 2023-06-14 02:30 AM - bill-auger

i added flashrom to the blacklist, with flashrom-stable as its replacement, for the 'technical' reason, and this ticket as the blacklist reference - i have not remade your-freedom though; so currently both packages are in the repos

from the mailing list:

as it stands, we decided that flashrom-stable would not be a duplicate package, but can replace flashrom instead - maybe that needs more discussion

if one of them (and only one of them) is deficient, then we should probably ignore the deficient one, regardless of its maintenance benefits - if both are deficient (in terms of features) in some way, then maybe we should carry both, or decide which one has the most desirable feature-set

generally, such decision should benefit users - concerns such as "it is too rusty" or "it is a PITA to maintain" do not affect the user; so those should be secondary concerns, not a deciding factor, unless the decision would not affect the user

### #2 - 2023-06-14 02:48 AM - bill-auger

WRT the pkgdesc (also from the mailing list), that is one of the reasons why i drilled wael for details - if both packages are to be kept, the pkgdesc should probably distinguish them somehow; but not so verbosely as proposed on the mailing list

```
- pkgdesc="Flashrom is a utility which can be used to detect, read, erase, or write BIOS chips (DIP, PLCC, SPI)."
```

```
+ pkgdesc="Flashrom is a utility which can be used to detect, read, erase, or write BIOS chips (DIP, PLCC, SPI)."  
+ pkgdesc+=" This flashrom-stalbe version doesn't require rust, so it is easier to build"  
+ pkgdesc+=" and it still builds fine on the same platforms supported by flashrom v1.2"
```

that proposed pkgdesc conflicts with the arch guidelines for two reasons:

- This is recommended to be 80 characters or less
- and should not include the package name in a self-referencing way

so, i have been trying to determine how that should look, in brief terms - is there any way to distinguish the two packages in say, less than 5 words?

the upstream pkgdesc is already pushing the limit of verbosity (>80 characters) - such implementation or maintenance details (how we build it, or why - eg: "doesn't require rust") are not features of the binary package - anything not relevant to users of the binary package, seems out-of-place in the pkgdesc - pkgdesc is intended to be very brief (one line in a shell) for `pacman -Ss` - wael added those rationales to the PKGBUILD