

## Packages - Housekeeping #2494

### [arrayfire] not supported by Arch Linux 32

2019-10-01 02:26 AM - Megver83

<b>Status:</b> open	<b>% Done:</b> 0%
<b>Priority:</b> bug	
<b>Assignee:</b> oaken-source	
<b>Category:</b>	
<b>Description</b>	
I was cleaning the blacklist from old packages and realized that arrayfire is only supported by upstream Arch Linux (x86_64 only), and Arch Linux 32 removed it because arrayfire devs are not interested in i686 (that's why they blacklisted it). Should we remove arrayfire for i686 and only support it for x86_64?	
<a href="https://git.archlinux32.org/archlinux32/packages/src/branch/master/blacklist/pentium4/arrayfire/arrayfire">https://git.archlinux32.org/archlinux32/packages/src/branch/master/blacklist/pentium4/arrayfire/arrayfire</a>	
I'm assigning this to oaken-source since he's the last packager	

### History

#### #1 - 2019-10-01 03:51 AM - bill-auger

i suppose that depends on what you meant by "devs are not interested in i686" - do you mean that the arrayfire upstream does not want to support i686? - thats becoming more common - mozilla and anything with rust are not giving much love to any 32bit arches these days - arch32 keeps trying to build that stuff though - if arch32 dropped arrayfire from i686, then probably we can too - thats probably is because it doesnt build anymore

note though in that URL, 'pentium4' is actually a different arch than i686 - arch32 has 3 arches now 486, i686, and pentium4 - roughly speaking: no-SSE, SSE, and SSE2

oaken-source has not been working on parabola for the past few months; so he may not respond

#### #2 - 2019-10-01 08:48 AM - oaken-source

don't worry, I'm still here. I'm just very much stretched for time at the moment.

but I'm still reading mail.

I remember having trouble building arrayfire on arm and i686 before, in these cases I generally try and keep the package alive for all arches, unless it becomes too hard to maintain. Once that line is crossed, there are two options to consider:

- A) keep an outdated version in the repos
- B) get rid of it.

which ones of these we should choose depends on the popularity of the package on that arch, if it is still widely used, we should keep it, otherwise it can go. The difficulty of course is judging whether a package is widely used or not.

I would expect situations like this to come up more frequently the less popular true 32 bit machines become, and the more developers of software drop support as a consequence. It might make sense to formulate a couple of guidelines.

#### #3 - 2019-10-01 11:35 PM - bill-auger

the first thing i want to make a point of is that these sort of "what should we do?" discussions should be on the mailing list for the benefit of everyone, hackers and users alike - a bug tracker is for tracking progress of work to be done; such as a 'housekeeping' task, which is something that has already been decided should be done

almost every important discussion this year has been on the IRC or redmine; and the mailing list is nearly unused - that is really not good in terms of transparency - anyone new who is considering to use parabola, and wondering how "active" development is, would draw the very inaccurate conclusion from

viewing the mailing list archives, that either nothing is happening or that development discussions are private, whereby users are invited to inform development decisions only in the form of bug reports and specific feature requests

users and transparency aside, it is an unfortunate circumstance that i have noticed lately, that some of the official 'hackers' have little or no communication with the team - some rarely or never join the IRC channel, are not reading the mailing list, keeping their GPG keys valid, or even keeping a valid email address

the mailing list is ideal for that basic level of transparent team communication, requiring the least amount of effort to stay "in the loop" with emerging discussions and infrastructure changes, for those who are not doing the work, but are still interested - the new forum can replace the assist list FWIW; but there are still many people subscribed to the dev mailing list - no one who is interested in following parabola development is going to subscribe to the bug tracker for that purpose

admittedly, i have not minded this concern myself as much as i could have - here is one of several things that only myself and freemor have discussed recently - the hackers.git entries are not well correlated with redmine and parabolaweb anymore - recently, i have started correlating these; which is still a tedious manual process - there are several people with the 'hacker' role and shell access that have not been active for a very long time and i am planning to move them to 'fellows' soon - thats not to say that contributing tangible work is the essential criteria; but anyone considered to be "on the team" should be in some communication with the team; at least reading the mailing list, and willing to add their input to important discussions routinely

if we are not routinely using the mailing list for discussions, then it is very difficult to know which 'hackers' have any interest in parabola at all - probably oaken-source, aurelein, fauno, isacdaavid, and lukeshu are in that category; but the others i am planning to demote to 'fellows' and revoke shell access, after i make a list of any packages they may have in the system and re-package them

#### **#4 - 2019-10-01 11:58 PM - bill-auger**

back to the topic: id say that there already is a guideline for this situation - the mission statement says that we will stay as close as possible to the upstream arch - o/c having multiple upstreams was not within the original scope of parabola; but arch32 and archarm have become the surrogate upstream arch for those repos - so without any more thought, we could just conclude that arch32 and archarm are the canonical upstreams for those arches; and we should track them in the same way as arch64 - i.e. anything that is not in arch32 does not need a libre replacement in [libre]; and would be entirely optional in [pcr]

if there are any unclear guidelines to discuss regarding this, i would suggest that everything in [libre] should be a replacement of something upstream, with an dedicated entry in the blacklist and corresponding bug report for future reference of the "why is this here?" question - if something like this is removed upstream, but we want to keep it; it should be removed from the blacklist and moved to [pcr], to make it clear that this package is no longer part of the standard distribution ([core], [extra], [community], [libre], [nonprosm], [nonsystemd]) - i.e. any package for which the answer to the "why is this here?" question is: "only because it was deemed to be useful, but the upstream does not carry it", should be in [pcr], for the sake of clearly semantic guidelines

using that distinction (the same distinction as upstream arch vs AUR), then it becomes immediately clear that when something can be removed from the blacklist, that it should be removed from [libre] at the same time - after all, the [libre] repo makes sense only because there are packages on the blacklist;

otherwise it could have any arbitrary name, (e.g. [parabola-extra]) - but that is essentially what [pcr] already is

with that clear semantic separation, then this becomes only a question of: "how valuable is that package after the upstream has dropped it?" (e.g. does anything valuable in [pcr] depend on it?, would anyone complain if it went away?) - thats not even a new criteria - it is the very same criteria as for any new packaging request for something to enter [pcr]; namely: the ( desirability / ( auditing\_cost + maintenance\_cost ) ) ratio

and again, if there is something more to discuss to reach a consensus regarding general guidelines, i think that deserves a mailing list thread

specifically for this package, in order to determine the conclusion to oaken-source's A+B propositions, i think the relevant questions are:

- A) did arch32 drop this package from i686 or only pentium4?
- B) what is the (desirability / cost) ratio for keeping it?

**#5 - 2019-10-10 01:05 AM - Megver83**

bill-auger wrote:

specifically for this package, in order to determine the conclusion to oaken-source's A+B propositions, i think the relevant questions are:

- A) did arch32 drop this package from i686 or only pentium4?
- B) what is the (desirability / cost) ratio for keeping it?

tl;dr but got your point

well I found out that Arch32 doesn't have arrayfire in its repos (for none of the arch'es they support), but they put it in their pentium4 blacklist (don't ask me why)

and regarding the guidelines for deciding what to do with packages... idk, I just don't see that there are like a bunch of users interested in the topic, but could be proposed of course in the forum

**#6 - 2019-10-10 01:12 AM - bill-auger**

if arch32 blacklisted it, that is very likely to be done because it wont build anyways

**#7 - 2019-10-10 01:30 AM - freemor**

The entries in the blacklist usually have a clue.

in this case it was:

```
"We neither test/build on 32bit architecture nor claim 32bit support."
```

Which I'm guessing is the response they got back from the arrayfire Devs