

## Re: Handling of shell builtins in make(1)

*Source:* <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2005-05/0087.html>

---

*From:* Harti Brandt ([hartmut.brandt\\_at\\_dlr.de](mailto:hartmut.brandt_at_dlr.de))

*Date:* 05/24/05

Date: Tue, 24 May 2005 10:28:30 +0200 (CEST)

To: Stijn Hoop <[stijn@win.tue.nl](mailto:stijn@win.tue.nl)>

On Tue, 24 May 2005, Stijn Hoop wrote:

SH>On Tue, May 24, 2005 at 10:08:19AM +0200, Harti Brandt wrote:

SH>> On Mon, 23 May 2005, Max Okumoto wrote:

SH>>

SH>> MO>4. Extend .SHELL: to allow specifying a list of built-in. And then

SH>> MO> we should define them in `bsd.sys.mk` or some other config file.

SH>>

SH>> We would need to add yet another config file for this because make

SH>> doesn't read anything if given the `-r` flag. It would end up having

SH>> no shell which would be wrong.

SH>

SH>Or, add a way to query `/bin/sh` for the list of builtins (`sh`

SH>`---show-builtins`) and use that in the absence of an explicit `.SHELL`.

SH>

SH>> MO>This will allow people to add keywords for their shells, and

SH>> MO>remove that stuff from the make source code. Hard coding it

SH>> MO>in the binary is wrong.

SH>>

SH>> The initial shell (`/bin/sh`) must be in the binary because it must run with

SH>> `-r`. I see no sense to add another config file just for the purpose

SH>> of specifying the shell. What would that give you that you cannot do now?

SH>

SH>I think it's more that hardcoding a list of builtins is undesirable.

That's no problem. We can just hardcode an empty list of builtins. In that case make always uses the shell. We can then put the actual list of builtins into `sys.mk` (or even retrieve it there from the shell via `$(/bin/sh ---show-builtins:sh)`). The question is do we really want to have, for example, 'bind' execute the shell builtin or do we want to execute it a bind somewhere in the path? From the point of makefile portability the later is desirable, because someone could legally have a script called bind in the build directory and call it via make. 'bind' is not in the list of builtins specified by Posix (I hope) so one would not expect to get any problems. On the other hand executing the bind builtin may be less surprising to people writing a makefile specifically for FreeBSD. That way, at an early stage, one may decide not to use bind as the name of a

freebsd-arch: Re: Handling of shell builtins in make(1)

script.

harti

---

freebsd-arch@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-arch>

To unsubscribe, send any mail to "freebsd-arch-unsubscribe@freebsd.org"