
Subject: Re: 2022(?).2 beta

Posted by [mirek](#) on Mon, 19 Dec 2022 22:35:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

Lance wrote on Mon, 19 December 2022 23:14mirek wrote on Mon, 19 December 2022 14:22Lance wrote on Mon, 19 December 2022 19:10And some of the viable options if multi-c++-version support is a necessity (as proposed by Klugier and me):

1. [=] to [&] when necessary. Make local copies of variables that are originally captured by value with undesired modification, and refer only to the copy in the lambda body. A fictitious example:

[&] does not help and the problem is not local copies.

This does not work:

```
struct MyApp : TopWindow {
    Button b;

    MyApp() {
        int j = 12;
        b << [&] { PromptOK(AsString(j)); };
    }
};
```

I see. Reference to local variables invalidated out of function body. So this option is eliminated. We are left with only 2.

3. Disable warning and hope that it will be deprecated for really long time. I bet it will.

See, this whole thing is rather unfortunate. There are 3 options, none of them really good. 2 of these require significant work and a chance of introducing new bugs....
