

Feedback re Go 2 Error Handling Draft Design

This page is meant to collect and organize feedback and discussion started with the Go 2 [error handling draft design](#).

Please post feedback on your blog, Medium, GitHub Gists, mailing lists, Google Docs, etc. And then please link it here.

Please help categorize the rest of the uncategorized proposals at the bottom.

Update, June 19, 2019 – Based on this feedback we have posted a [proposal for a try builtin](#), which produced significant feedback on #32437 ([threaded view here](#)).

We've disabled comments on that issue until July 1 to give people time to experiment in their own code. We would very much like to see experience reports about using the general try approach in your own code, including trial conversions with [tryhard](#). Thanks!

Experience with the [“try proposal”](#)

Add to this list:

- TEMPLATE: Your Name, [“Title”](#), Month YYYY
-

Requirements

Discussions of the requirements for a new error handling method.

- Warren Stephens, [“Go 2 handle should provide line number, source filename, version”](#), December 2018
- Liam Breck, [“Requirements to Consider for Go 2 Error Handling”](#), September 2018
- jimmyfrasche, [“Don't special case error or nil”](#), September 2018
- Matt Dee [“Error Handling Should Support Custom Error Types”](#), August 2018
- Ian Lance Taylor, [“Incomplete list of criteria”](#) from GitHub proposal discussion, May 2018
- Rob Pike (posted by @matjam) [“Simplicity is Complicated”](#), December 2015

In support

This includes supporting the existing chaining/stacking of handlers without changes.

- Tokyo Gophers, [“In support comments from Go 2 feedback event”](#), October 2018
- Adam Bouhenguel [“In support of more declarative error handling”](#), August 2018
- Daniel Theophanes, [“Go 2 Design: Error Handling Net Win”](#), August 2018
- Alexandru-Paul Copil, [“In Support, with slight modifications and considerations”](#), September 2021

Example code

Code changed to use the existing proposal.

- Daniel Milde, "[Use check-handle draft syntax in gdu](#)", May 2021
- Mateusz Czapliński, "[Converting a fragment of real code with error handling to Go 2 'design draft'](#)", August 2018
- Blake Mizerany, "[How best to account for partial writes when using check/handle?](#)", August 2018

Against

Critiques without counter-proposals

- Pasha Osipyants, "[Error handling mechanism using current go possibilities](#)", February 2019
- Anonymous, "[Go 2 Error Handling Non-Proposal](#)", October 2018
- Stripe developers, "[Feedback on Go 2 draft designs](#)", October 2018
- Tokyo Gophers, "[Against comments from Go 2 feedback event](#)", Oct 2018
- Liam Breck, "[Golang, How dare you handle my checks!](#)", September 2018
- Nate Finch, "[Handle and Check, Let's Not](#)", September 2018
- Jozef Slezak, "[Use semicolons instead of new keywords: check+handle](#)", September 2018
- Shannon Wynter "[Error Handling as it can already be done](#)", August 2018

Recurring themes

Concepts that appear repeatedly among the suggestions below.

- Invoke handler via assignment, e.g. `v, ? := f()`
references: [1](#), [2](#), [3](#), [4](#), [5](#), [6](#), [7](#), [8](#), [9](#), [10](#), [11](#), [12](#), [13](#), [14](#)
- Invoke one of several handlers by name, e.g. `v := check f() ? name`
references: [1](#), [2](#), [3](#), [4](#), [5](#), [6](#), [7](#), [8](#), [9](#), [10](#), [11](#), [12](#), [13](#), [14](#), [15](#), [16](#), [17](#)

Modest revisions

- Leidong Liu, "[a, b, !checkError := Sub\(..\)](#)", Nov 2019
- Jin Feng, "[A simplified Go 2 error handling solution with no handler], (<https://gist.github.com/jfeng45/416d4ecb42a4df50bebf86ce1c41c668>)", July 2019
- Dave Arnold, "[Move special check keyword to left-hand side of assignment](#)", April 2019
- Rockmen, "[Add handle and ? as syntax sugar maybe](#)", December 2018
- Steve Manuel, "[Go 2 handle statement to optionally use a type switch](#)", October 2018
- Zlatko Bratkovic, "[In support with tiny change](#)", October 2018

- DeedleFake, "[Possible Solution to `check` Awkwardness with Chained Method Calls](#)", September 2018
- Yesuu Zhang, "[Pass the check and handle parameters, custom handle](#)", September 2018
- Viktor Kojouharov, "[Reducing the special casing around the new error design draft](#)", September 2018
- Aleksei Pavliukov, "[Use a function as a handle parameter](#)", September 2018
- Savino Pio Liguori, "[Feedback for Go2 error handling design](#)", August 2018
- Jeffrey Koehler, "[In support of Handle Chaining: On Check](#)", August 2018
- Garrus, "[Another style of syntactic sugar on error handling](#)", August 2018
- Paul Borman, "[Arguments against the Go 2 error handling proposal](#)", August 2018
- krhubert, "[Error default handler for tests](#)", May 2019
- Franz, Implement error handling like an invariant check of a variable. when the value changes, a function is called. this can then be used in other situations too.
- Brandon Heenan, "[Concise, with more obvious control flow](#)", May 2021

Remove handler chaining

- Markus Heukelom, "[proposal: Improve error handling using `guard` and `must` keywords](#)", April 2019
- Alessandro Arzilli, "[Against check as an operator and handler chains](#)", August 2018
- Simon Howard, "[Go 2 errors response: One Handler Per Function](#)", August 2018
- Eli Bendersky, "[Thoughts on the Go 2 Error Handling proposal](#)", September 2018
- Yoshiki Shibukawa, "[Every handles should have return statement](#)", September 2018

Counter-proposals

Error handling with normal functions

- Azamat Kalberdiev, "[Handling every error with intuitive code](#)", March 2021
- Andrew Phillips, "[Use closures rather than handlers](#)", October 2018
- Taihei Morikuni, "[Use functions as an error handler, Add syntactic sugar to remove duplicated if statement](#)", September 2018
- Scott Pakin, "[Go 2 error handling based on non-local returns](#)", September 2018
- Greg Weber "[Error handling with functions and an error return?](#)", September 2018. Originally linked [gist](#).
- Gigi Sayfan, "[Go 2 error handling feedback + alternative solution](#)", September 2018
- Ruan Kunliang, "[Simple Error Handling for Go 2](#)", August 2018
- Martin Rode, "[The return of the return, Error Handling for Go 2](#)", November 2018

Labeled error handlers

- Joe Lapp, "[Local-only throw-catch error handling](#)", October 2021
- Danijel Premus, "[Use existing go labels](#)", December 2018
- Burak Serdar, "[Handler for err declares both err and errHandler, less intrusive labeled error handling](#)", October 2018
- John Forstmeier, "[Labeled error handling](#)", September 2018
- Mikaël Cluseau, "[Multiple handlers, unambiguous on which return value is used](#)", September 2018
- Kiura Magomadov, "[Addition to Go2 draft error handling](#)", September 2018
- Liam Breck, "[The #id/catch Error Model](#)", September 2018
- Marlon Che, "[How about separating check and handle?](#)", August 2018

Inlining

- Patrick Kelly, "[handling more than just errors in go](#)", August 2018
- Vlad Didenko, "[Error Handling with grab | name\(\)](#)", November 2017
- Goooid, "[Inline style error handle\(simple unambiguous\)](#)", August 2018

Use defer

- Victoria Raymond, "[Force 'check' to return error instead of allowing customized logic](#)", August 2018
- Night-walker and daokoder, "[Extend and repurpose defer instead of introducing new syntax](#)", June 2014

try/catch/finally syntax

- Mathieu Devos, "[Go2 Error Handling Proposal: Scoped Check/Handle](#)", August 2018
- Rust RFC, "[Trait-Based Exception Handling \(wherein catch is akin to try\)](#)", February 2016
- ZhiFeng Hu, "[\[Go2ErrorHandling\] Go should support Exception handler](#)", August 2018
- Jan Semmelink, "[if-else-undo-done](#)", August 2018
- Vladimir Utoplov, "[Handling throws/throw idiom](#)", September 2018
- Gokan EKINCI, "[try-with-resources](#)", December 2018

Other possibilities

- Alex Hornbake, "[refuse. conditional return keyword](#)", June 2019
- yaxinx, "[Use ? as suffix to remove the last error value from multi-value tuples](#)", March 2019
- Plamen Stoev, "[Go 2 block-level checks](#)", November 2018

- Peter Goetz, "[Formalize and Enforce Error Handling](#)", September 2017
- Fedir RYKHTIK, "[Go 2 error handling with exclamation mark](#)", September 2018
- Einthusan Vigneswaran, "[Error Aware Keywords - return, defer, if, != and forcing the error object to be the last argument](#)", September 2018
- Andrew Phillips, "[Improving Go Error Handling](#)", October 2017

Uncategorized

Please help categorize the rest of the proposals here.

- Joe Lapp, " [on...return for error handling](#)", Oct 2021
- Andrew Gwozdziwycz, "[Check for Go Errors](#)", June 2019
- DeedleFake, "[Feedback for Go 2 Design Drafts](#)", August 2018
- Loki Verloren, "[Go 2 error handling feedback and my thoughts on how to improve programmer's efficiency and experience](#)", September 2018
- Gima, "[Procedural code, separate error handling](#)", June 2018
- Konstantin, "[error handling for error-tree](#)", [Discysson](#) May 2019

Adding your feedback

Please format all entries as below.

- *Your Name*, "[Title](#)", month year

To make it easier to see new feedback, please add your new proposal to the top of the section it is placed in.