

build passing

build passing

go reference

go report A+

errors

* used by 16.2k projects

Package errors provides simple error handling primitives.

```
go get github.com/pkg/errors
```

The traditional error handling idiom in Go is roughly akin to

```
if err != nil {  
    return err  
}
```

which applied recursively up the call stack results in error reports without context or debugging information. The errors package allows programmers to add context to the failure path in their code in a way that does not destroy the original value of the error.

Adding context to an error

The errors.Wrap function returns a new error that adds context to the original error. For example

```
_, err := ioutil.ReadAll(r)  
if err != nil {  
    return errors.Wrap(err, "read failed")  
}
```

Retrieving the cause of an error

Using errors.Wrap constructs a stack of errors, adding context to the preceding error. Depending on the nature of the error it may be necessary to reverse the operation of errors.Wrap to retrieve the original error for inspection. Any error value which implements this interface can be inspected by errors.Cause .

```
type causer interface {  
    Cause() error  
}
```

errors.Cause will recursively retrieve the topmost error which does not implement causer , which is assumed to be the original cause. For example:

```
switch err := errors.Cause(err).(type) {  
case *MyError:  
    // handle specifically  
default:
```

```
// unknown error  
}
```

[Read the package documentation for more information.](#)

Roadmap

With the upcoming [Go2 error proposals](#) this package is moving into maintenance mode. The roadmap for a 1.0 release is as follows:

- 0.9. Remove pre Go 1.9 and Go 1.10 support, address outstanding pull requests (if possible)
- 1.0. Final release.

Contributing

Because of the Go2 errors changes, this package is not accepting proposals for new functionality. With that said, we welcome pull requests, bug fixes and issue reports.

Before sending a PR, please discuss your change by raising an issue.

License

BSD-2-Clause