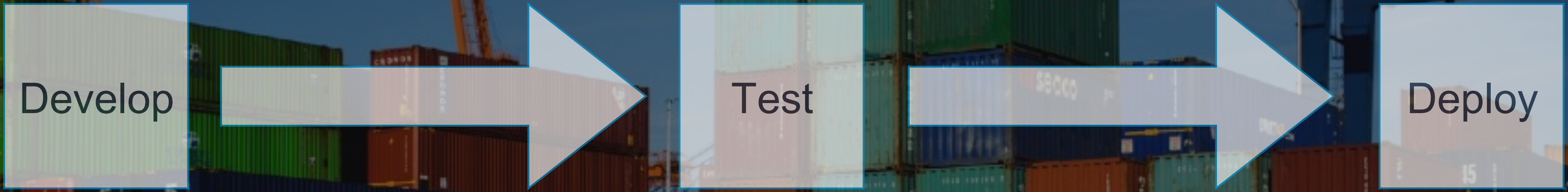
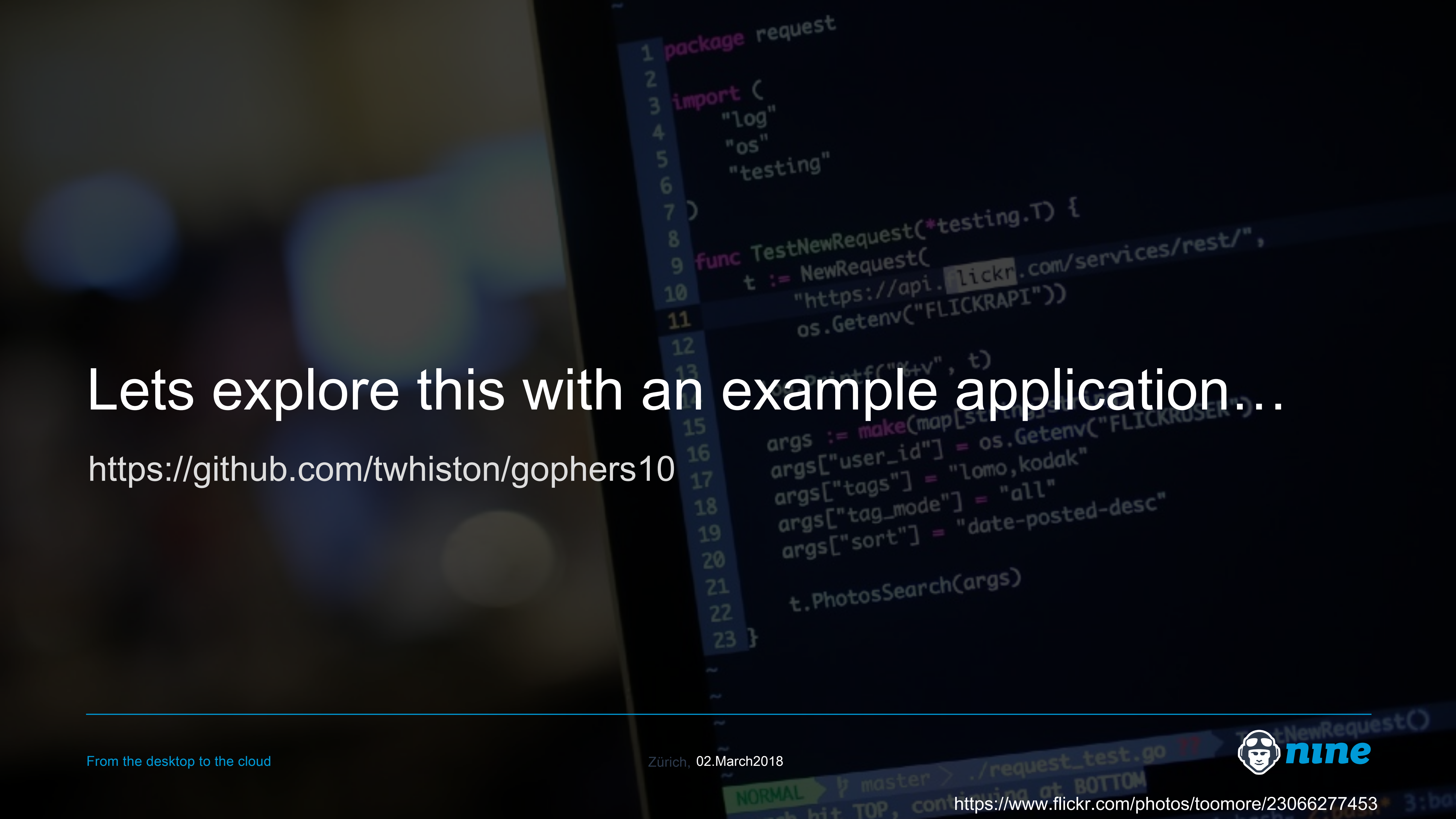
How do we manage this complexity across environment boundaries?



- Containerising the application provides a stable and reproducible environment
- Containerisation is the ‘native’ format of the cloud
- Reduces complexity around scaling
- However introduces increased complexity around tooling

Lets explore this with an example application...

https://github.com/twhiston/gophers10



```
package request

import (
	"log"
	"os"
	"testing"
)

func TestNewRequest(*testing.T) {
	t := NewRequest(
		"https://api.flickr.com/services/rest/",
		os.Getenv("FLICKRAPI"))

	Printf("%+v", t)

	args := make(map[st
	args["user_id"] = os.Getenv("FLICKRUSER")
	args["tags"] = "lomo,kodak"
	args["tag_mode"] = "all"
	args["sort"] = "date-posted-desc"

	t.PhotosSearch(args)
}
```

https://www.flickr.com/photos/toomore/23066277453