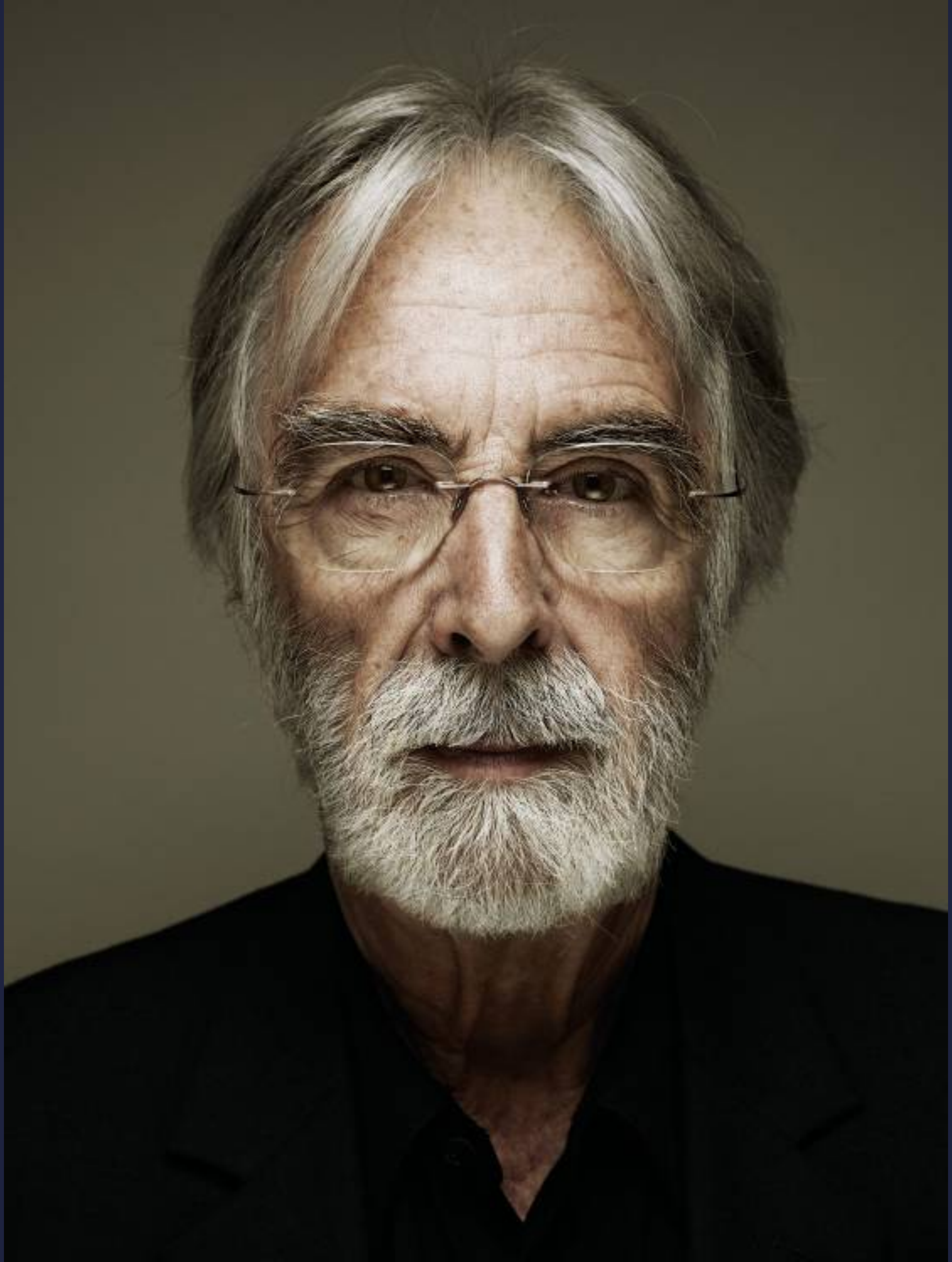
LEARNING SWIFT BY DEVELOPING HANEKE



github.com/Haneke

A lightweight zero-config image cache for iOS.

```
[imageView hnk_setImageFromURL:url];
```

- Asynchronous image retrieval
- First-level memory cache using NSCache
- Second-level LRU disk cache using the file system
- Background image resizing
- Image decompression
- Optimized for UITableView and UICollectionView cell reuse

HANEKE 1.0.0

- UIButton category
- Independent LRU disk cache
- Optional AFNetworking integration
- Audited APIs and documentation
- Fully unit tested (until proven otherwise)

AVAILABLE TODAY

@oriolblanc @joanromano @lascorbe @hpique

github.com/realm/summer-of-swift

HANEKE<T>

v0.1.0

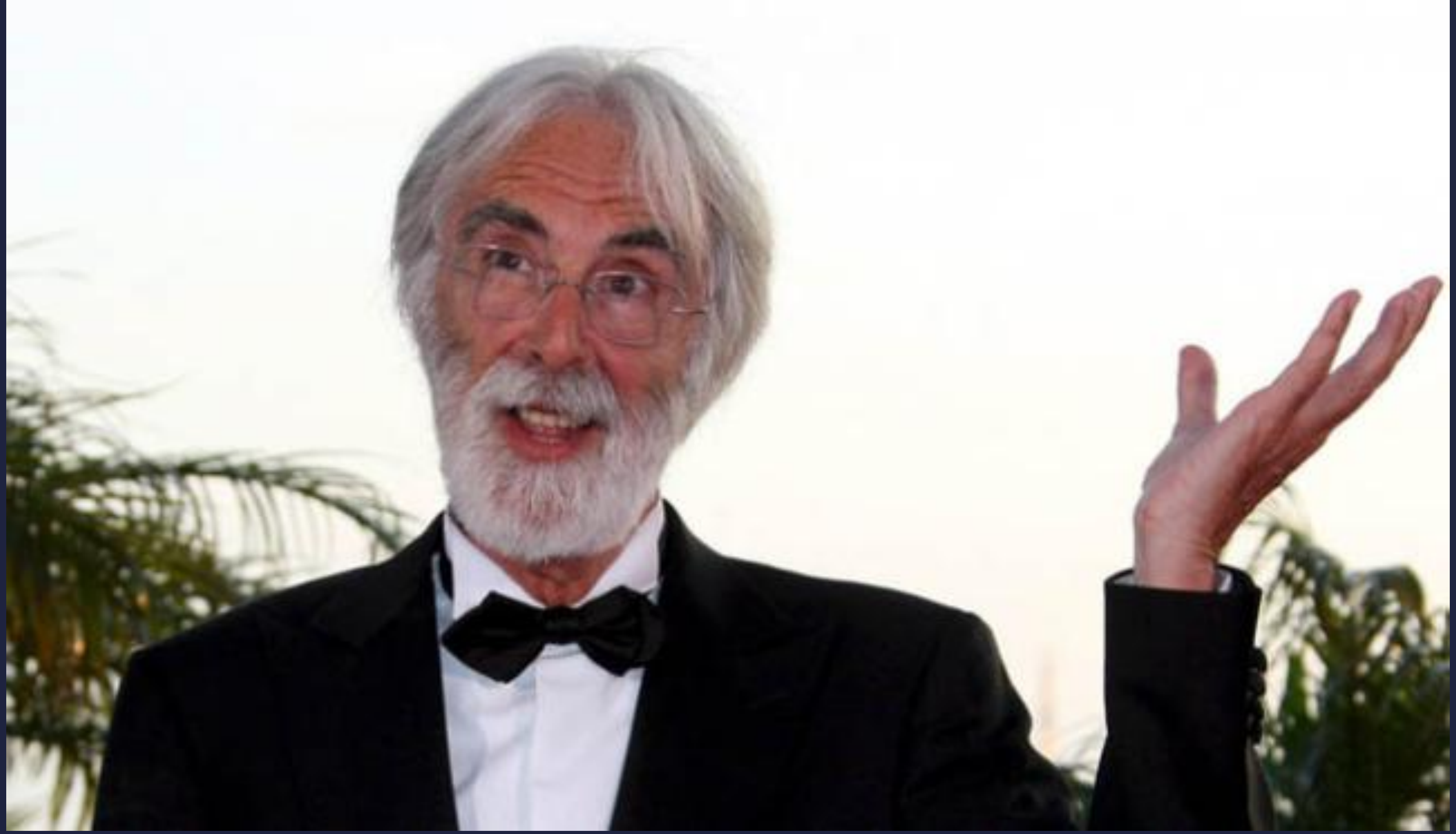
A lightweight generic cache for iOS, in Swift.

```
let cache = Cache<UIImage>("images")
```

With all the features of its Objective-C counterpart*

```
imageView.hnk_setImageFromURL(url)
```

```
let cache = Cache<UIImage>("images")
let cache = Cache<NSData>("data")
let cache = Cache<String>("text")
```



FEATURING

SQL GENERICS

```
class Cache<T : DataConvertible where T.Result == T> {...}

protocol DataConvertible {
    typealias Result
    class func convertFromData(data:NSData) -> Result?
}
```

stackoverflow.com/questions/25922152/not-identical-to-self

stackoverflow.com/questions/25915306/generic-closure-in-protocol

MIKE ASH CODE

```
func getKey() -> COpaquePointer {
    let ptr: COpaquePointer =
    Unmanaged<AnyObject>.passUnretained(self).toOpaque()
    return ptr
}
```

THE AMAZING UNDERSCORE

BAD

```
dispatch_sync(sut.cacheQueue, {
    fileManager.removeItemAtPath(path, error: &error)
})
```

GOOD

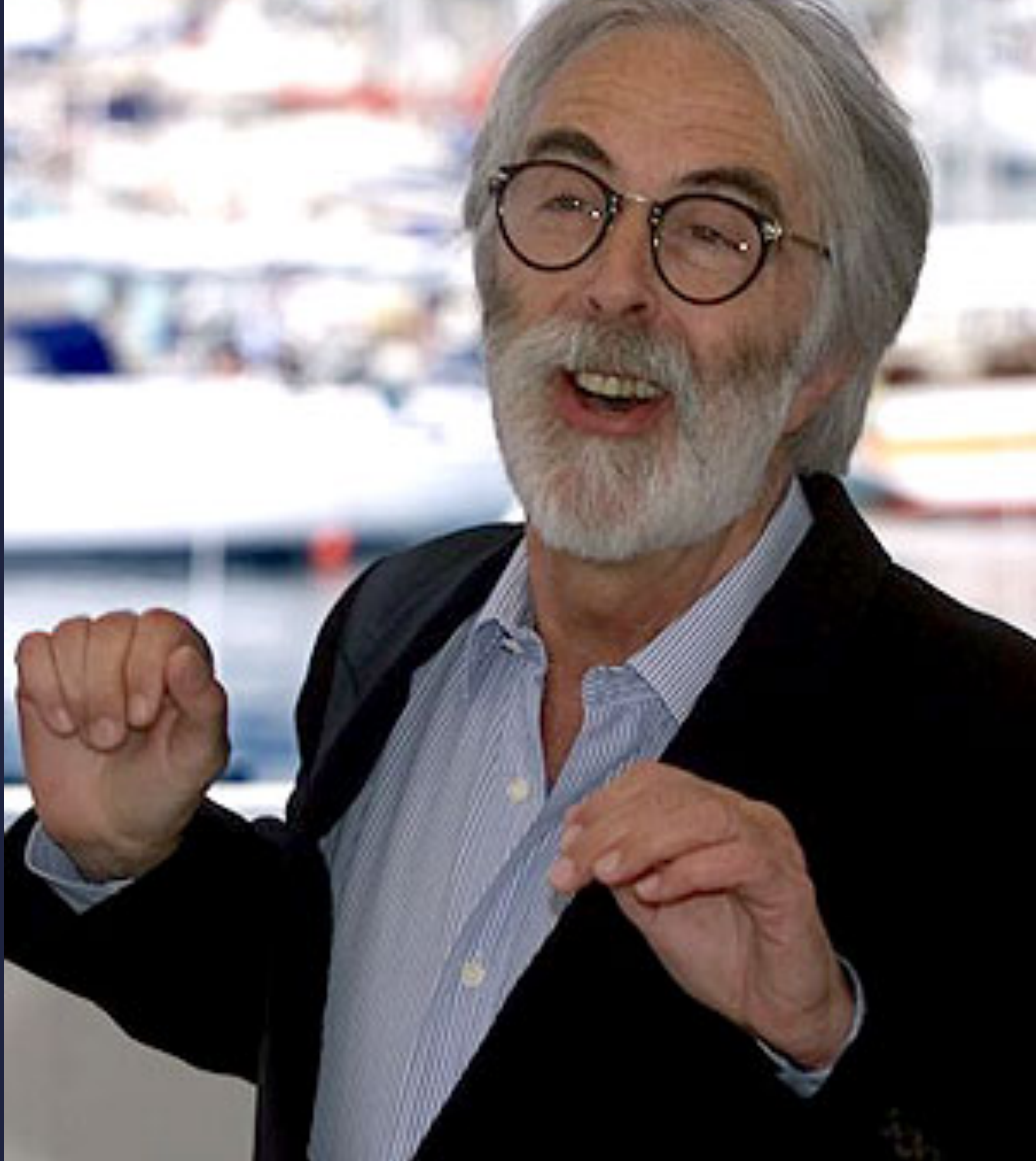
```
dispatch_sync(sut.cacheQueue, {
    let _ = fileManager.removeItemAtPath(path, error: &error)
})
```

MUTANT NINJA TUPLES

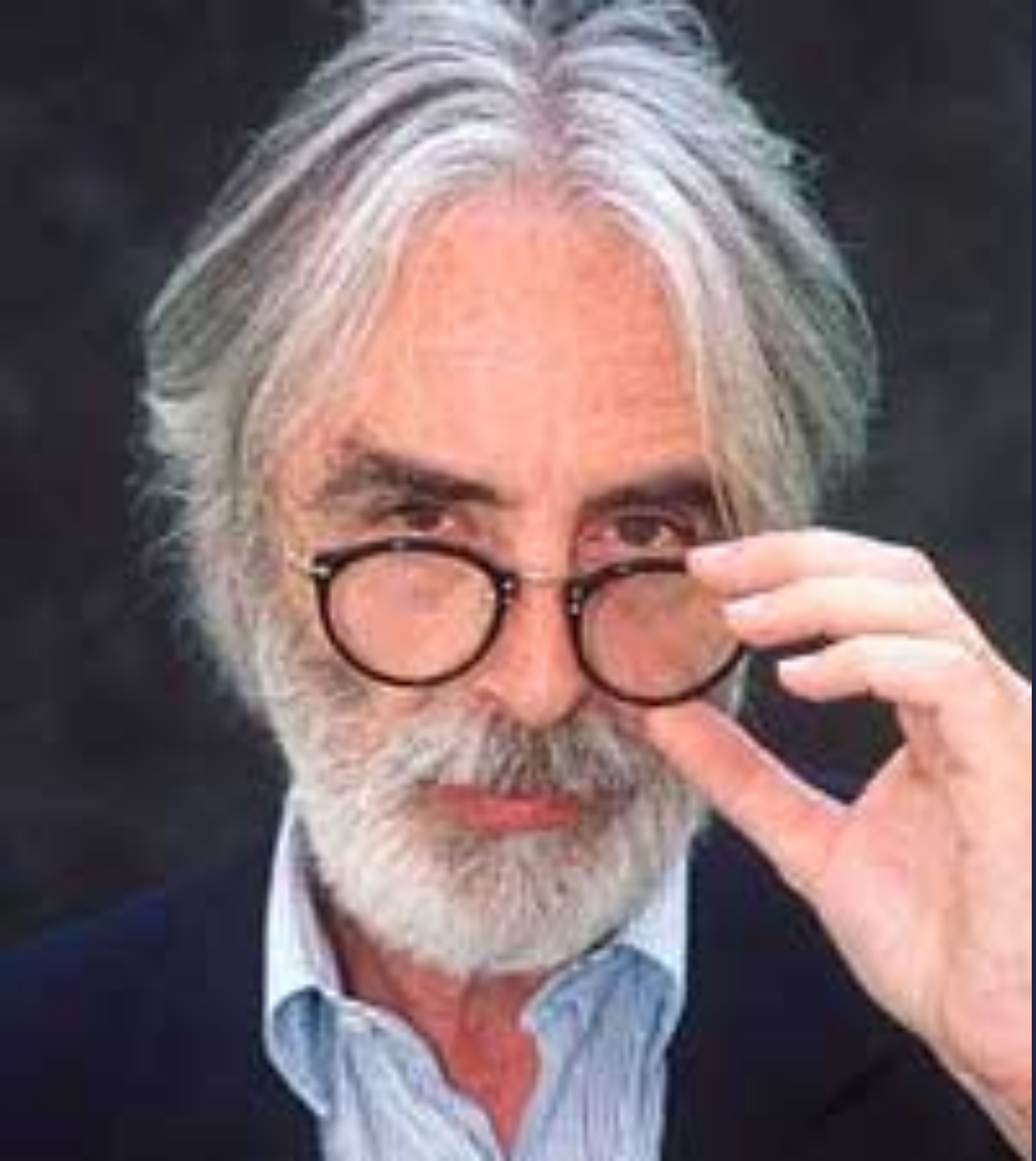
```
func onMemoryWarning() {
    for (_, (_, memoryCache, _)) in self.formats {
        memoryCache.removeAllObjects()
    }
}
```

AND MORE

- @autoclosures to optimize performance
- test expectations
- mocking with function classes
- bitwise operations
- structs as namespaces
- extensions
- interaction with C APIs



TAKEAWAYS



SWIFT IS NOT FINISHED

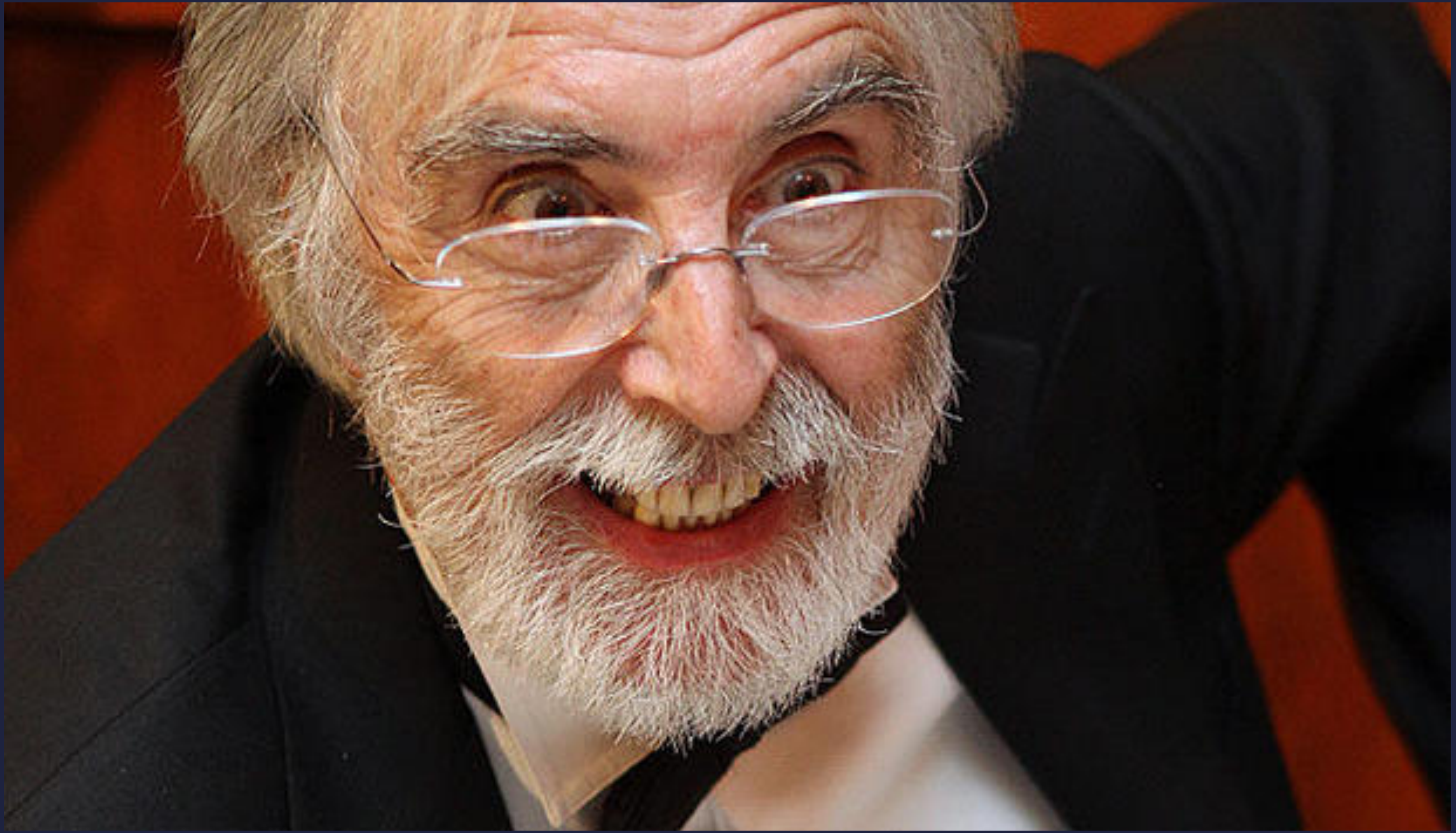
BAD

```
let image = UIImage(data: data)
```

GOOD

```
if let image : UIImage! = UIImage(data: data) { ... }
```

CHANGE BEYOND FUNCTIONAL



from

MUTABILITY

to

INMUTABILITY

from

CLASSES

to

ANY

from

VERBOSITY

to

INFERENCE

github.com/Haneke