

A few tips for threading with Core Data on iOS 5

Daniel Tull

Old style threading

NSPersistentStoreCoordinator

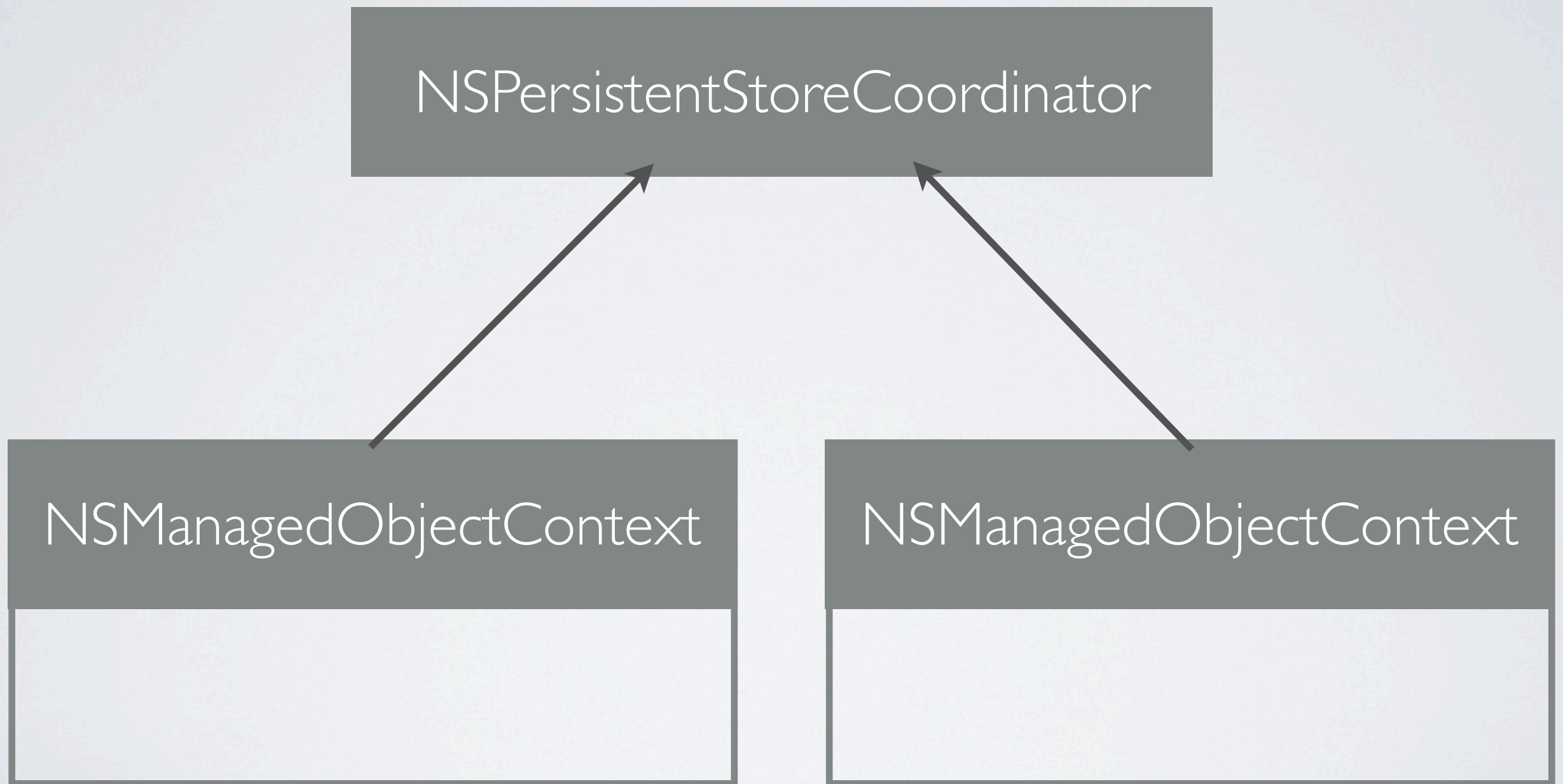
NSManagedObjectContext



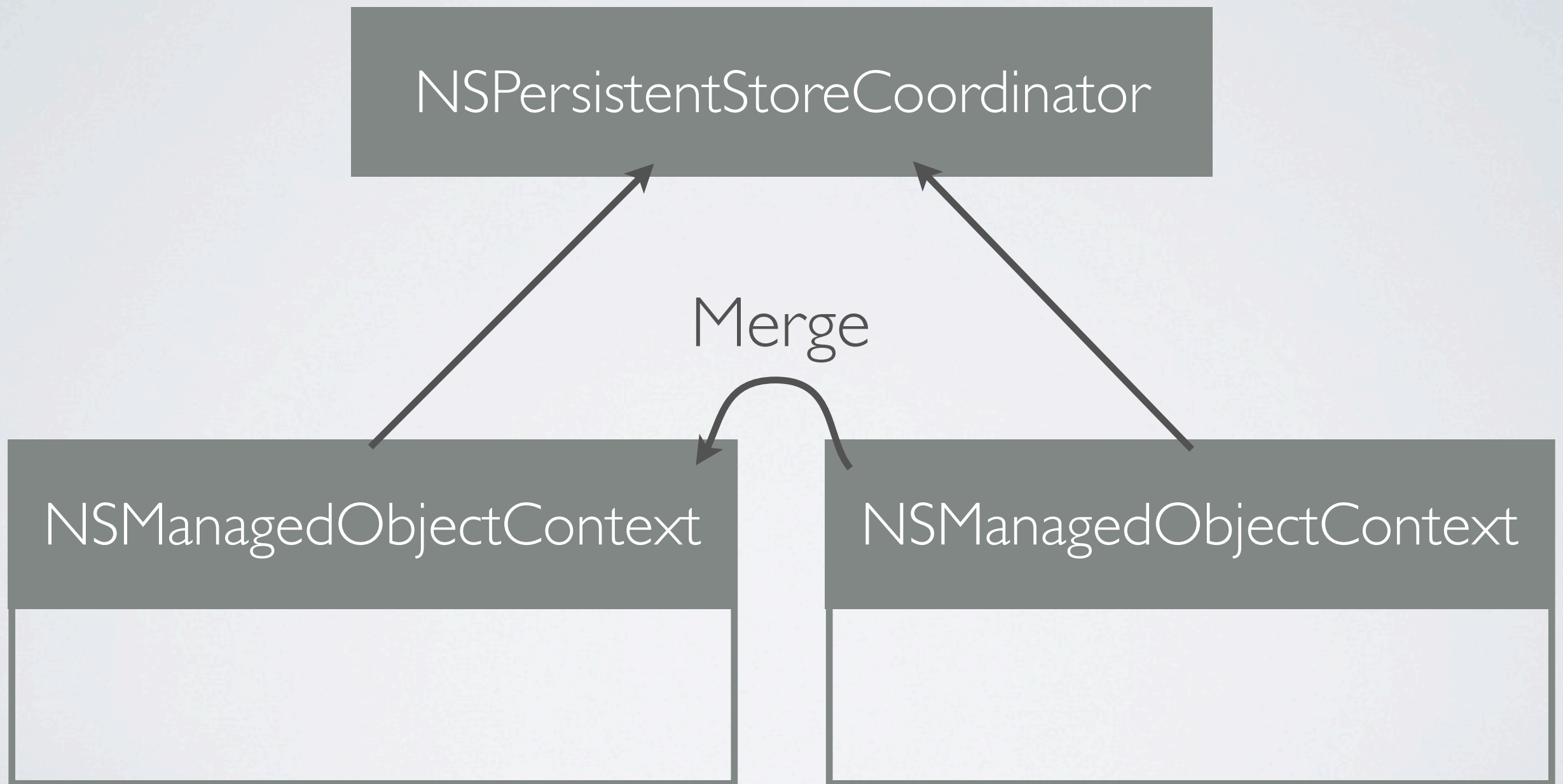
```
graph BT; A[NSManagedObjectContext] --> B[NSPersistentStoreCoordinator]
```

The diagram illustrates a relationship between two classes. A box labeled 'NSManagedObjectContext' is positioned at the bottom left. A box labeled 'NSPersistentStoreCoordinator' is positioned at the top right. A black arrow points from the top right corner of the 'NSManagedObjectContext' box to the bottom left corner of the 'NSPersistentStoreCoordinator' box, indicating a directed association or dependency.

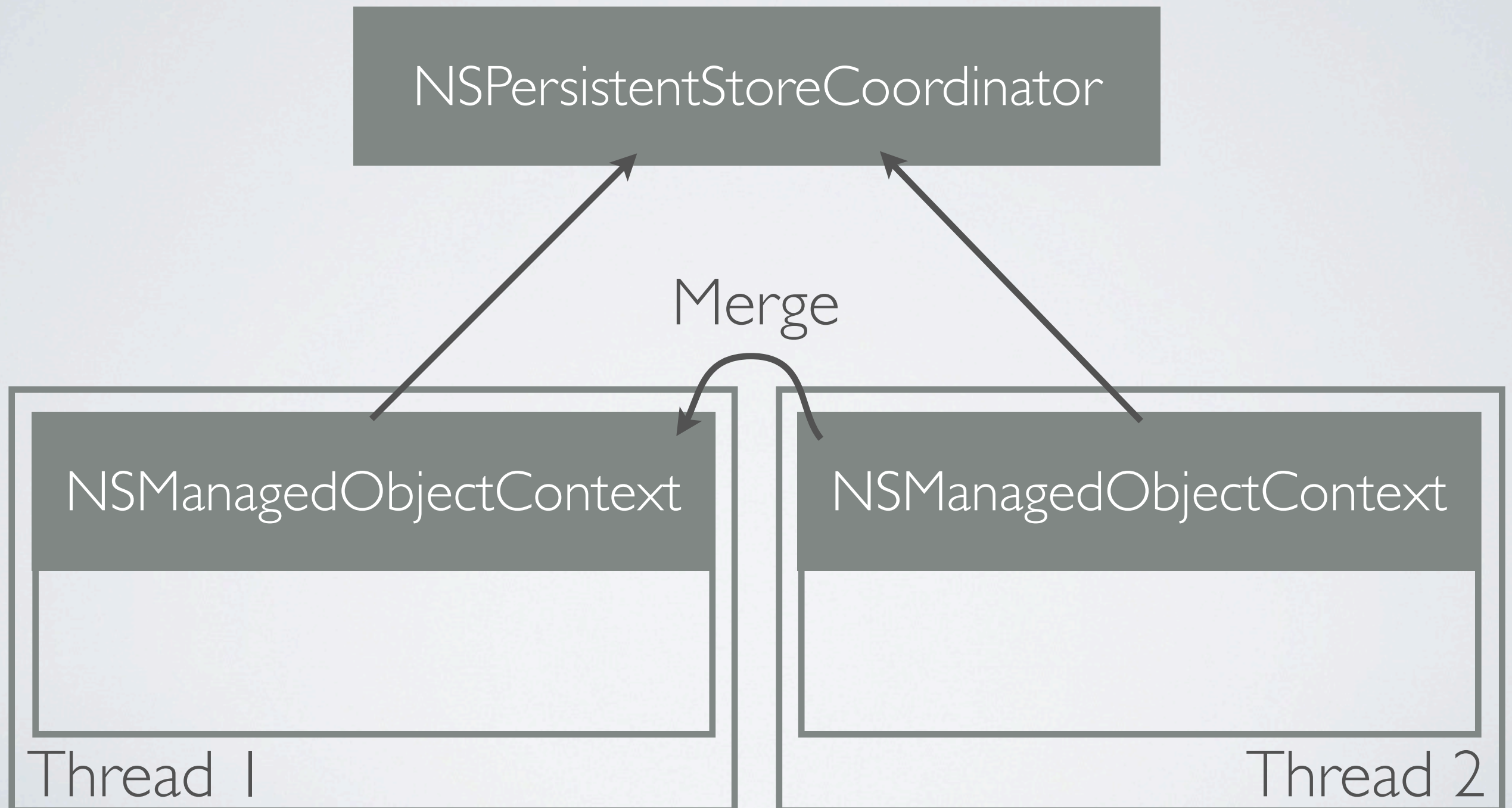
Old style threading



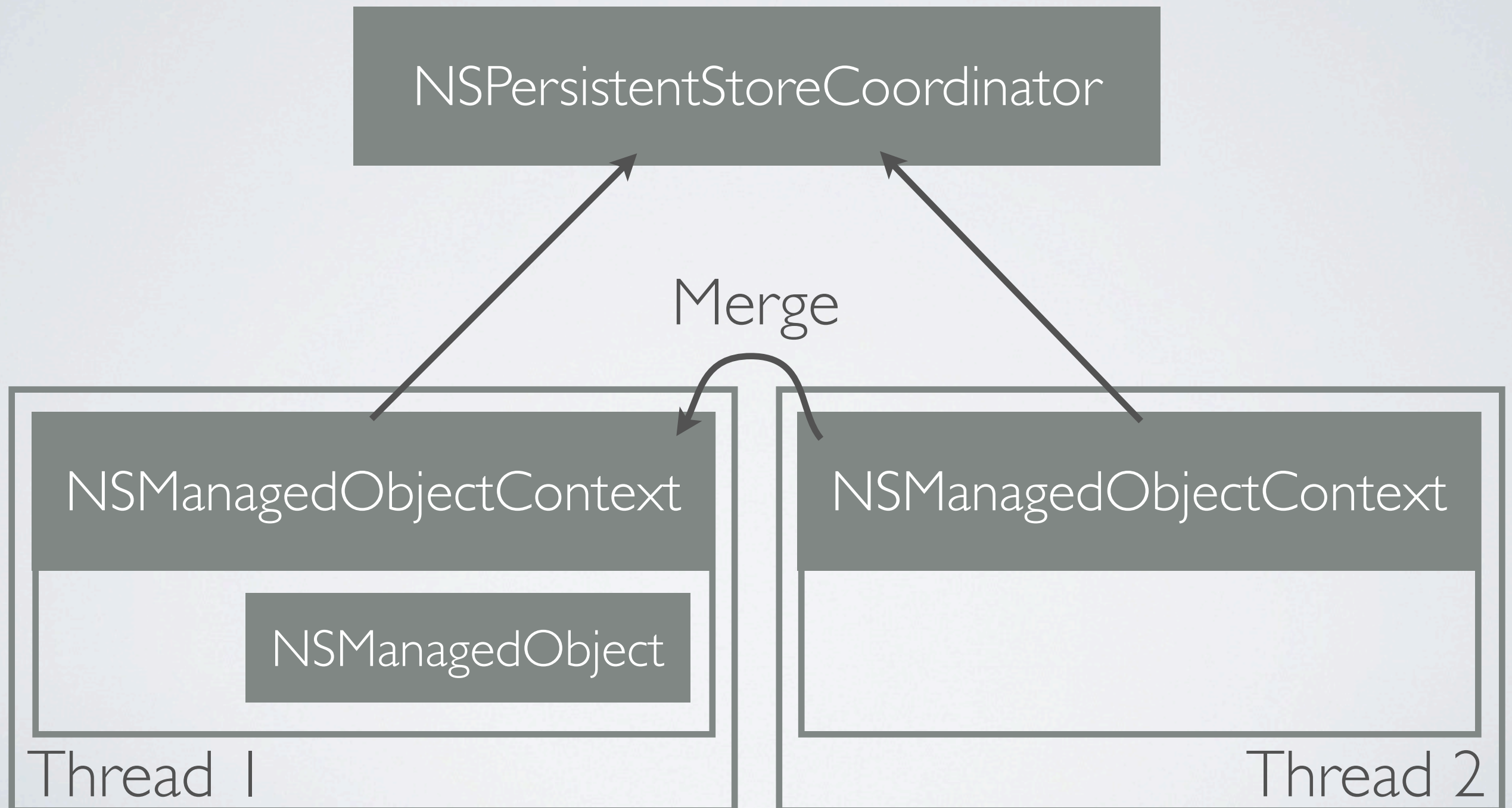
Old style threading



Old style threading

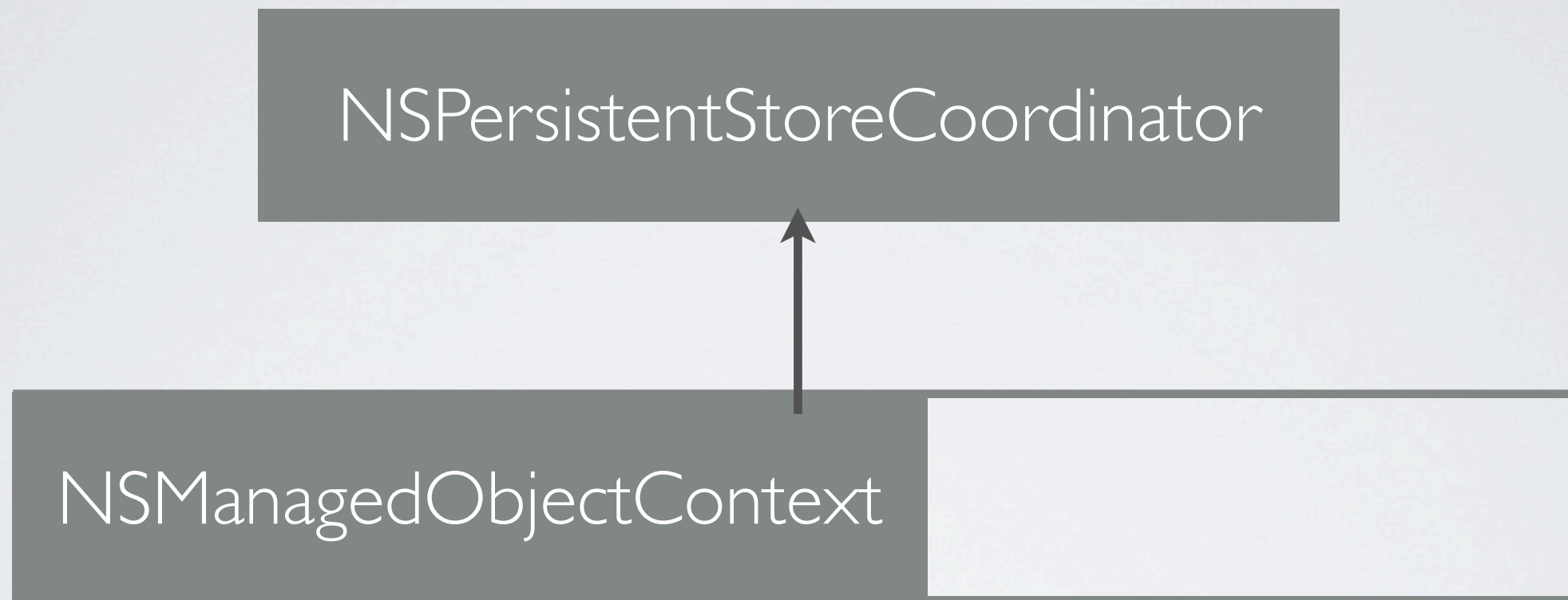


Old style threading

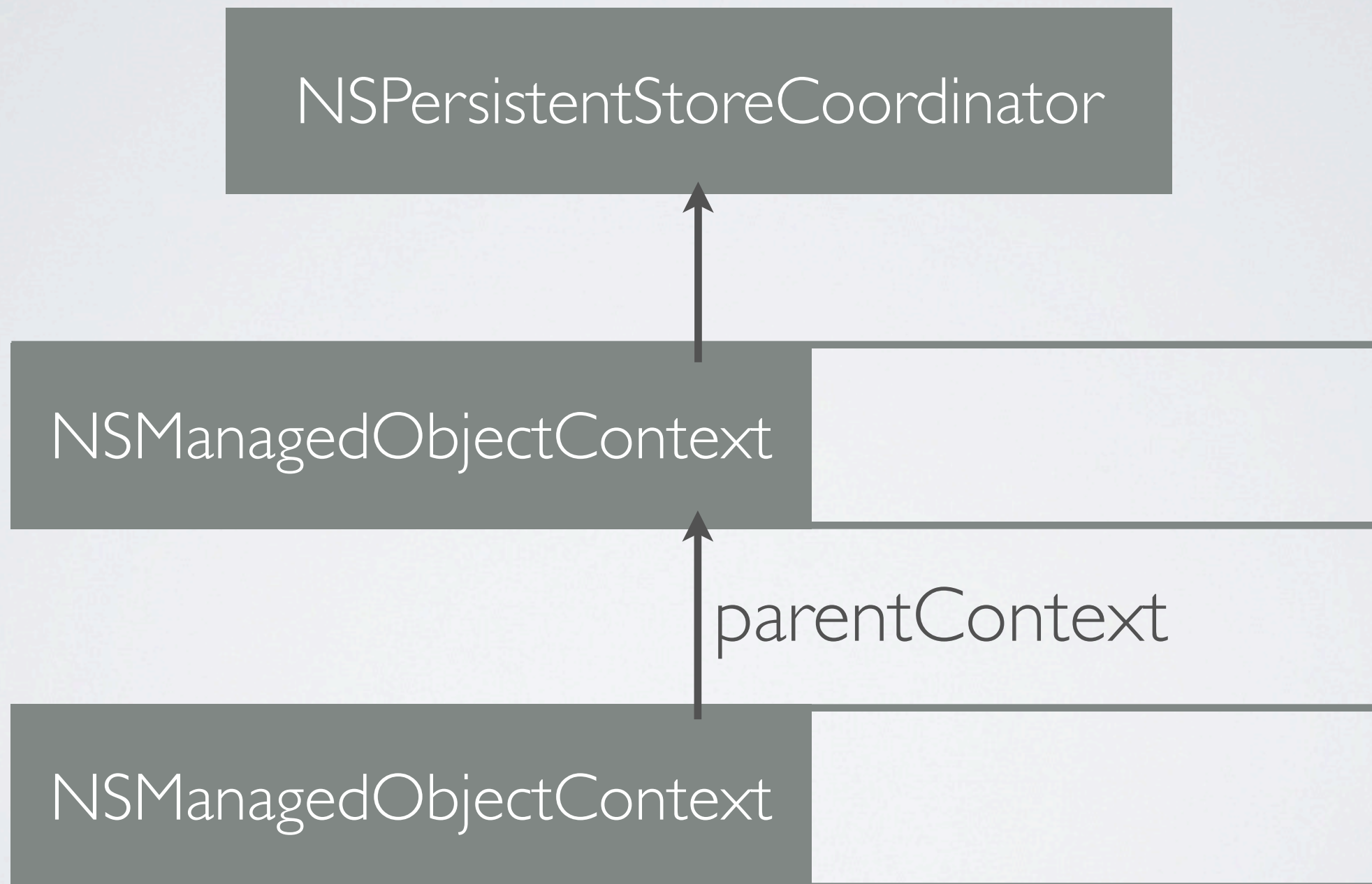


Old style threading

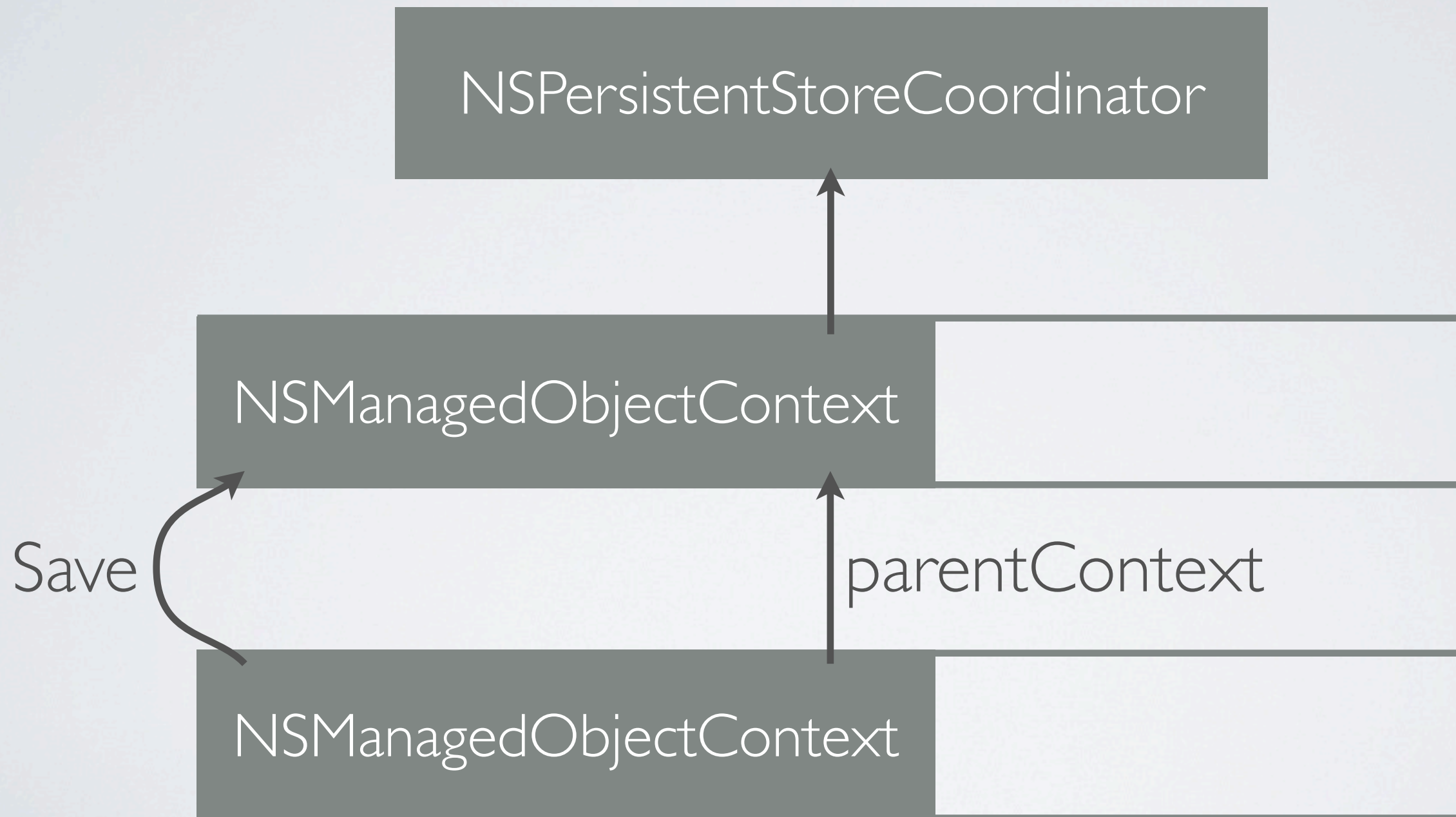
New style threading



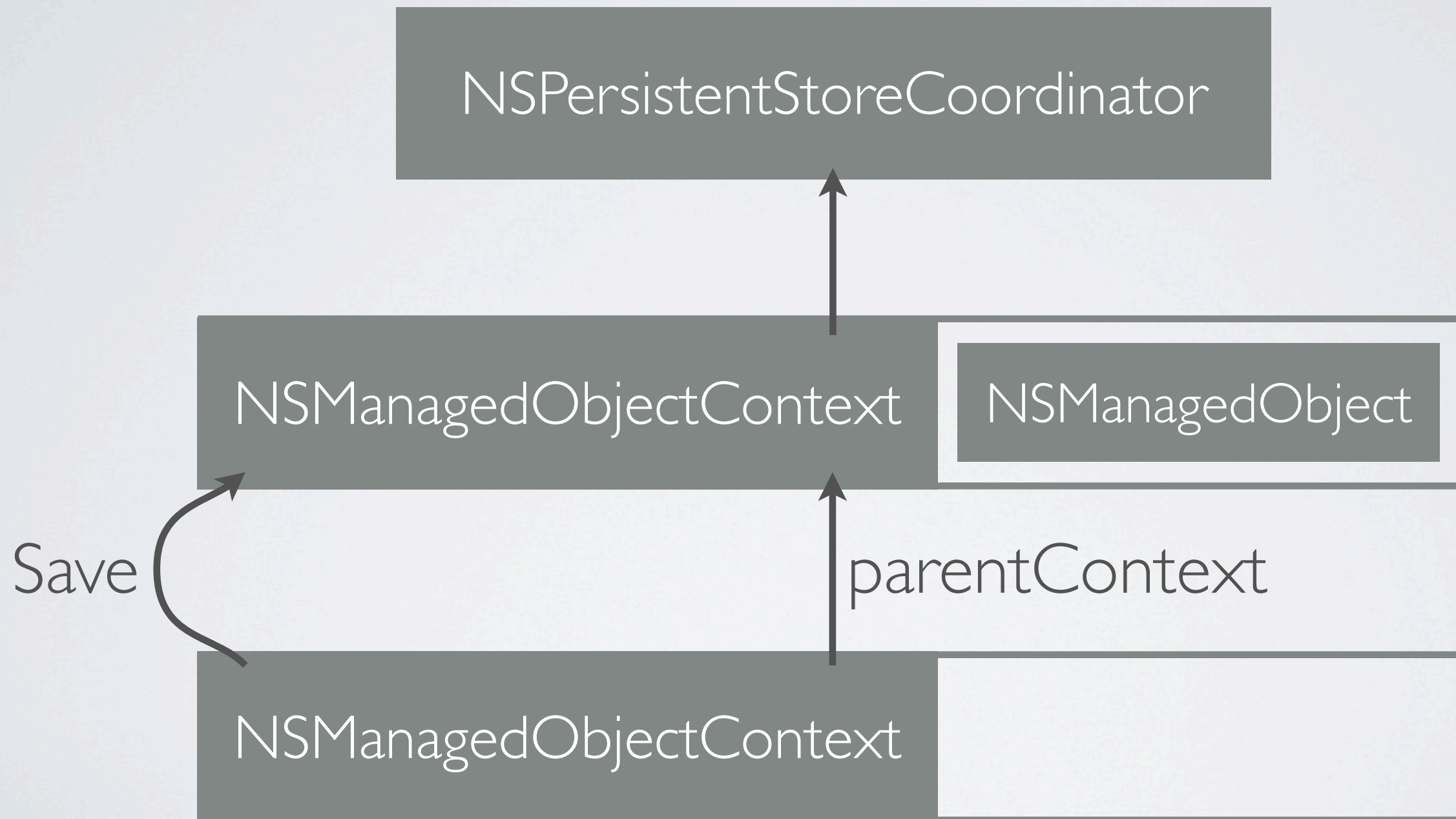
New style threading



New style threading



New style threading



New style threading

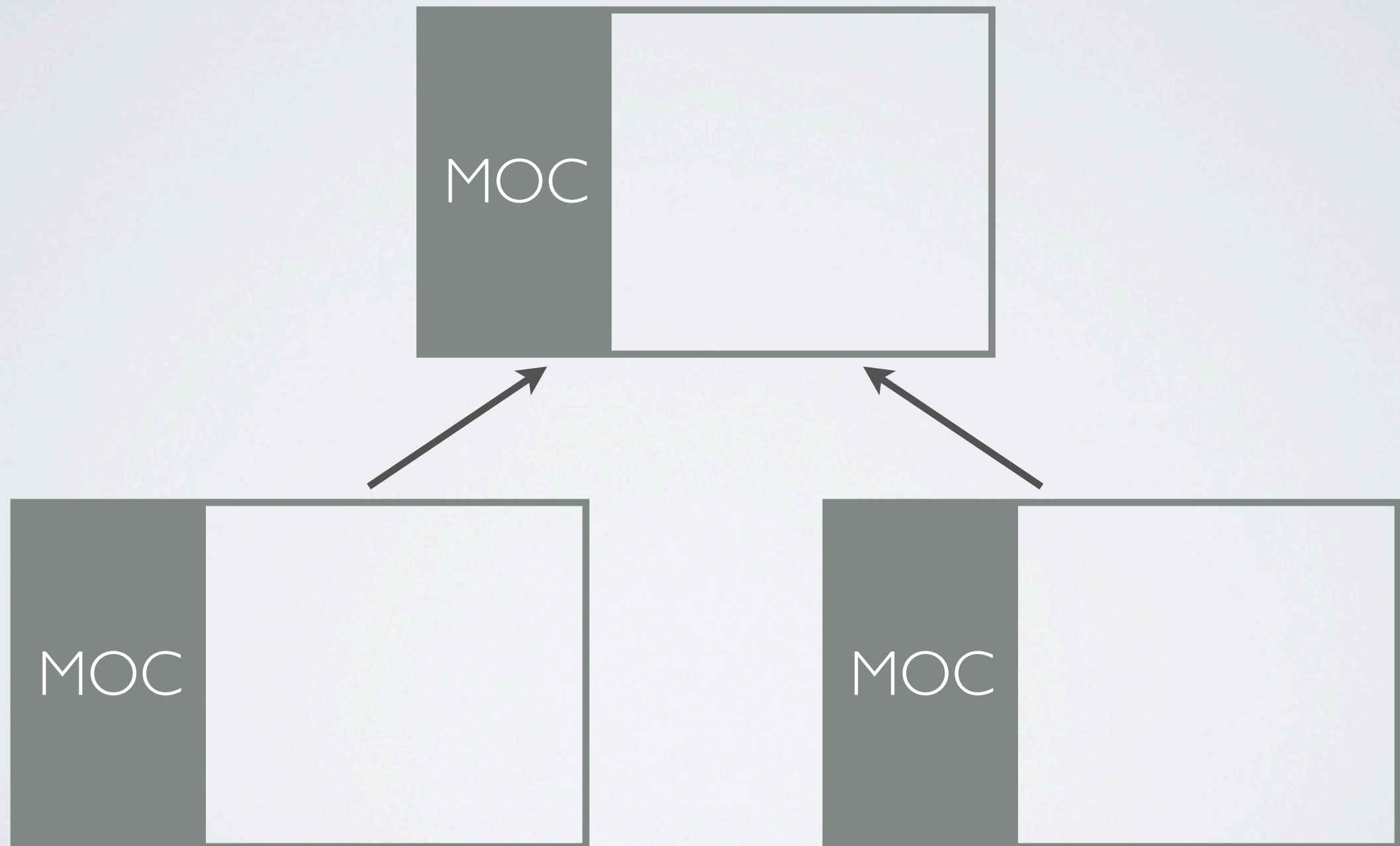
New style threading

Saving pushes the contents of the context up **one** level

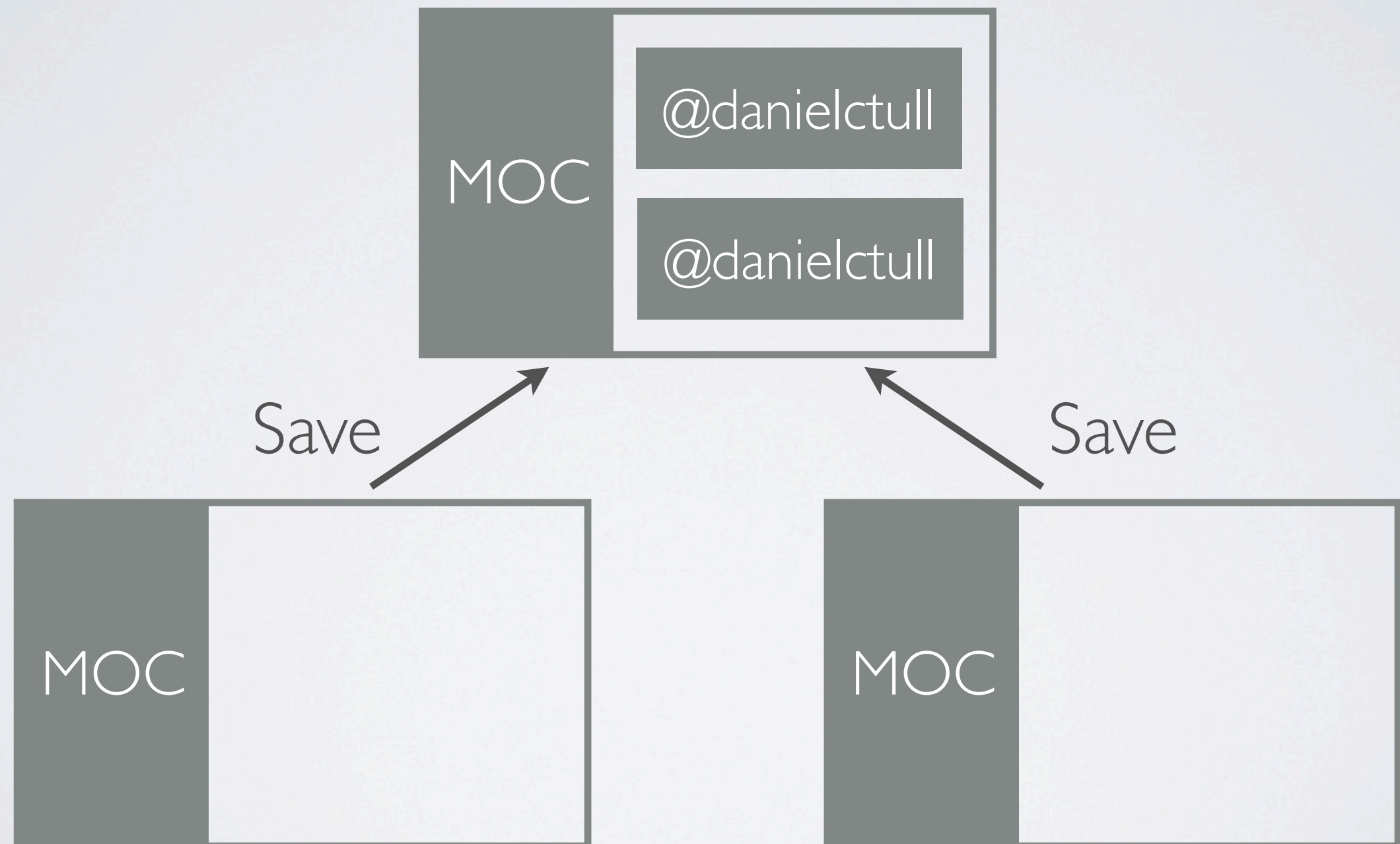
Still only one thread per managed object context

Made easier with internal threads (performBlock:)

Use one context for importing



Use one context for importing



Save asynchronously

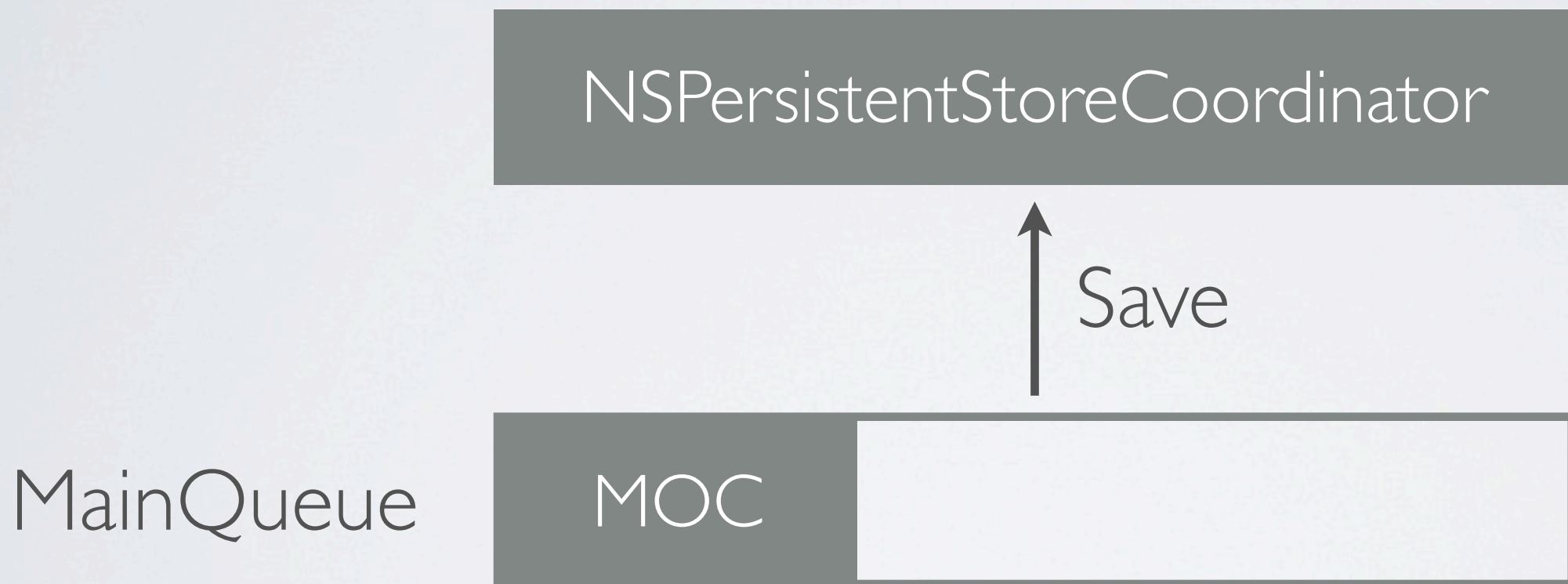
NSPersistentStoreCoordinator

MainQueue

MOC

@danielctull

Save asynchronously



Save asynchronously

NSPersistentStoreCoordinator

PrivateQueue

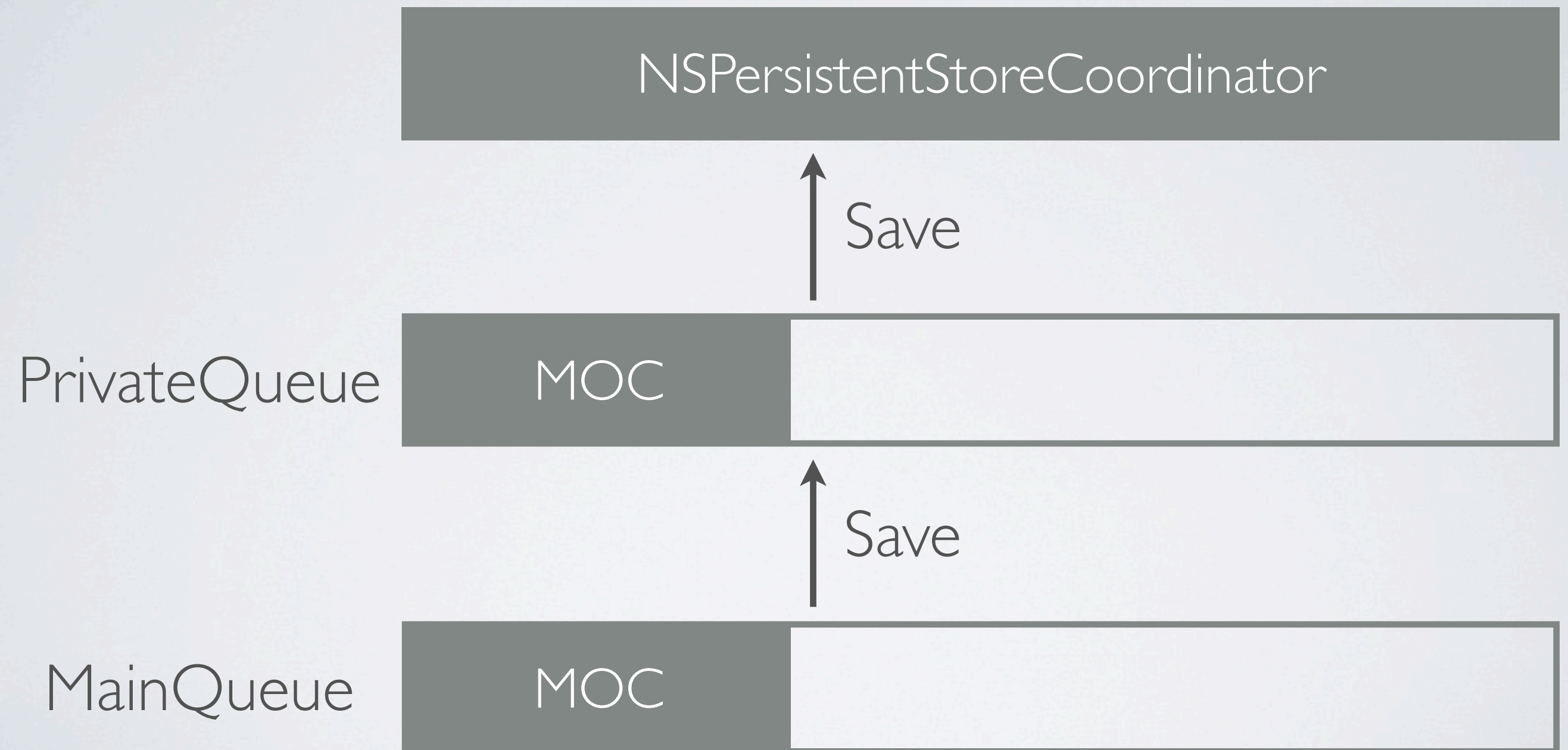
MOC

MainQueue

MOC

@danielctull

Save asynchronously

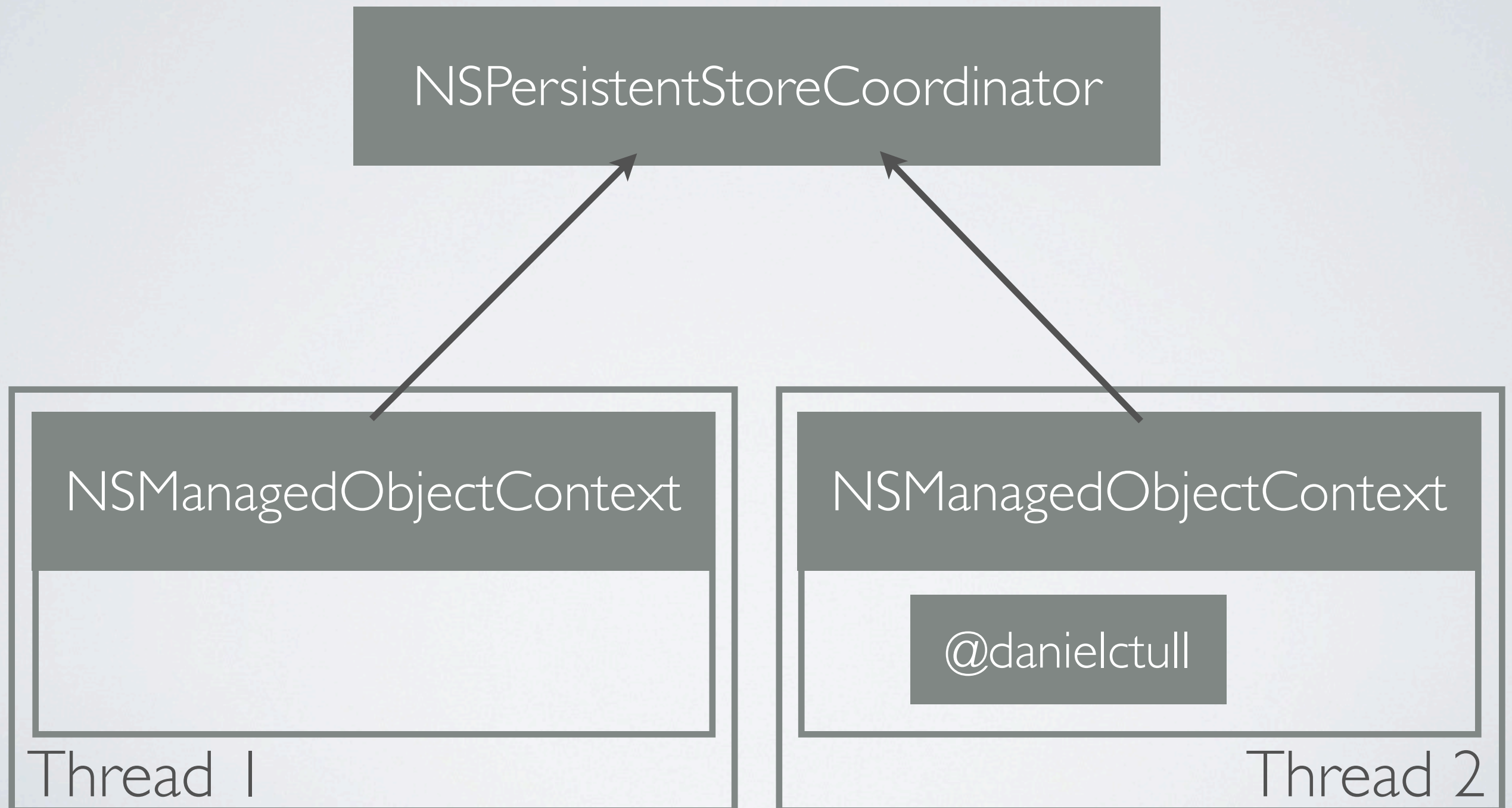


Get permanent object IDs

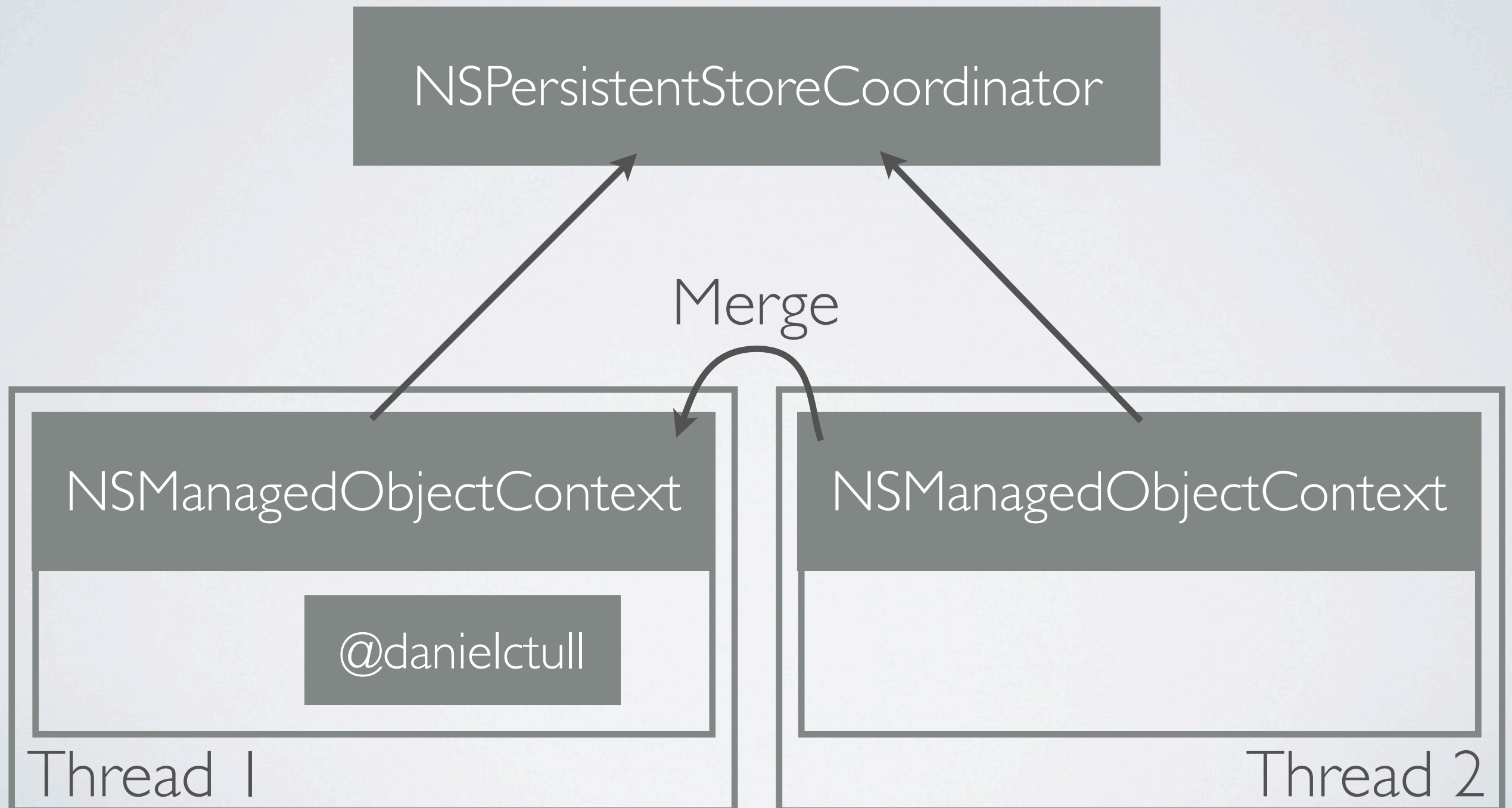
Certain requests use object ID to equate managed objects

If this is temporary, the request won't work

Get permanent object IDs



Get permanent object IDs



Save asynchronously

NSPersistentStoreCoordinator

MOC

MOC

@danielctull

Save asynchronously

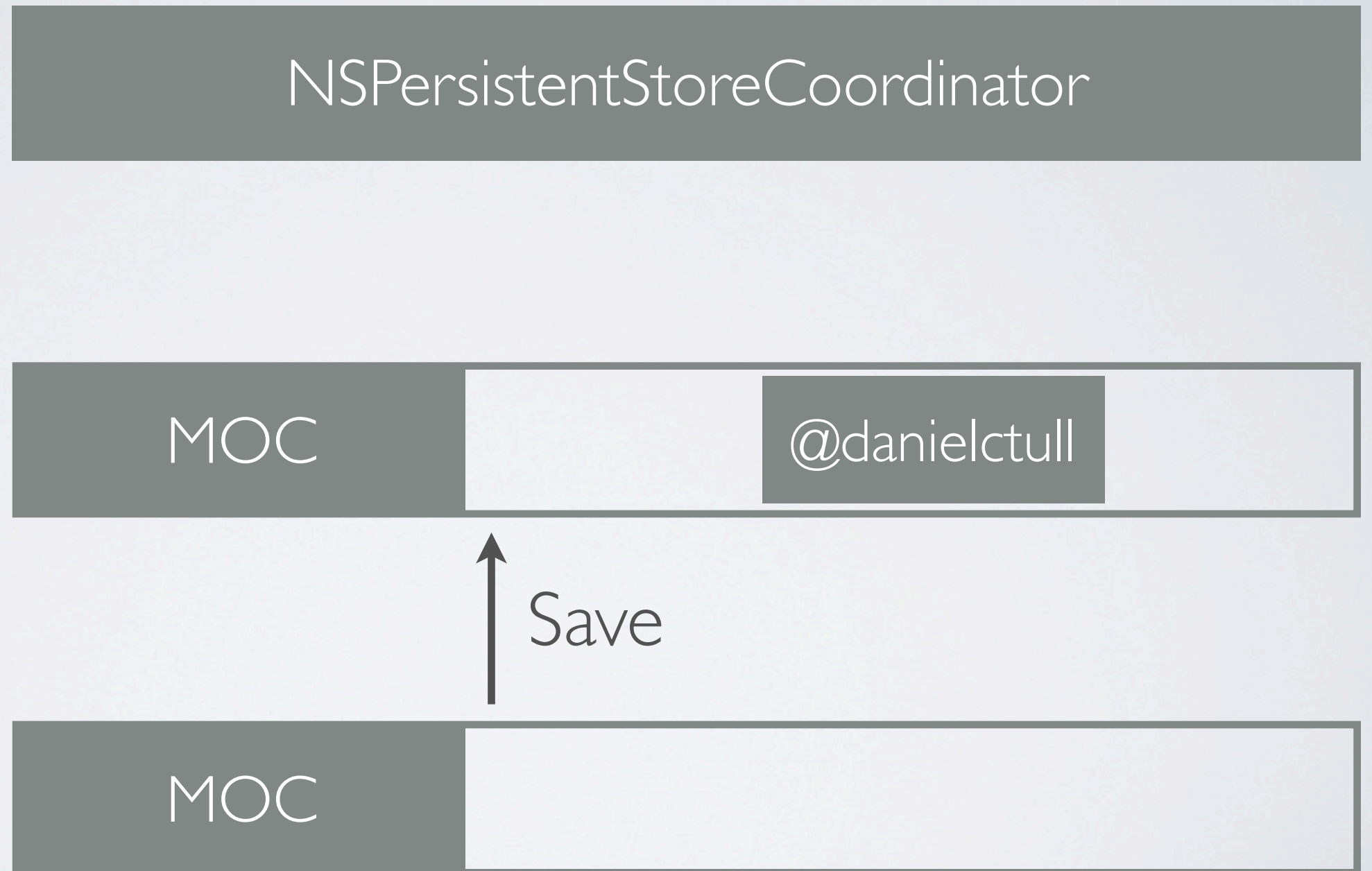
NSPersistentStoreCoordinator

MOC

@danielctull

Save

MOC



Get permanent object IDs

Object IDs aren't permanent until objects are saved to disk

or:

`-obtainPermanentIDsForObjects:error:`

Daniel Tull

@danielctull