

# Changing Core Data Models

---



**Andrew Bancroft**

@andrewcbancroft [www.andrewcbancroft.com](http://www.andrewcbancroft.com)

# Overview

Recently deployed ShoutOut to a pilot group of employees

Received ideas and suggestions from the testers

Some of the changes will affect the **data model**

- Create new data model version
- Migrate from old to new data model

Finalize the app by implementing the testers' feedback

# Creating Data Model Versions

---

# When to Create New Model Versions

*Very* early  
stages of app  
development

*Prior* to app  
being  
published in  
App store

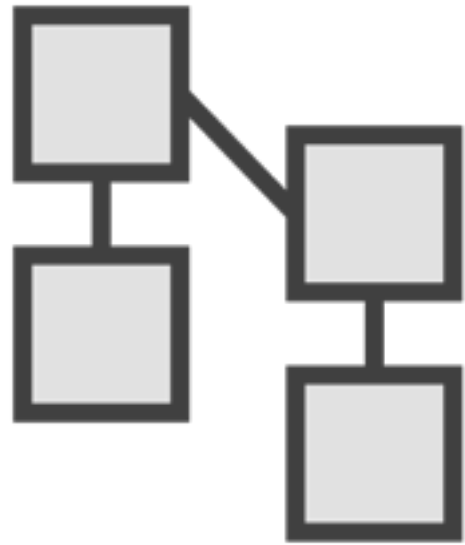
Handy when  
fleshing out  
*initial* data  
model

```
try! FileManager.default.removeItem(at: storeURL)
```

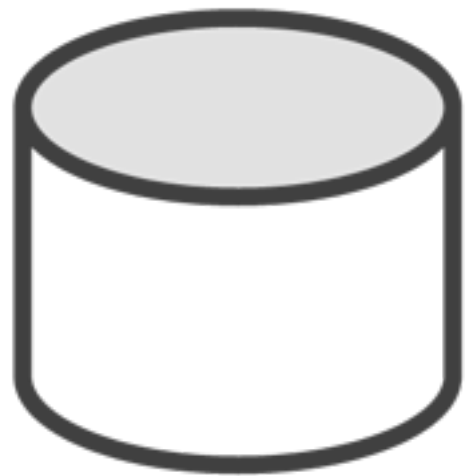
```
try! psc.addPersistentStore(  
    ofType: NSSQLiteStoreType,  
    configurationName: nil,  
    at: storeURL,  
    options: nil)
```

# When to Create New Model Versions





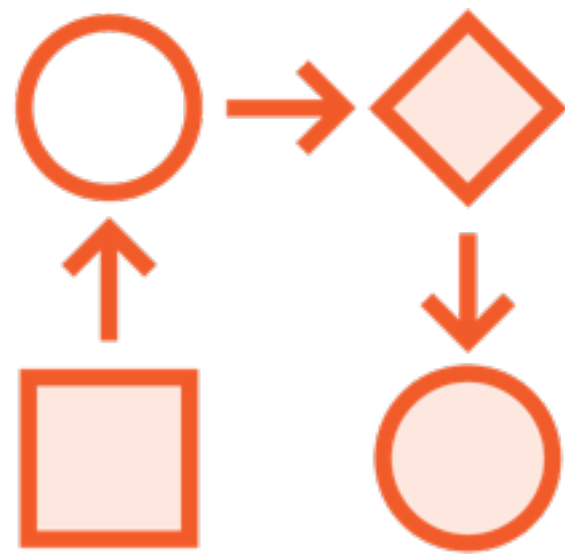
**Data Model**



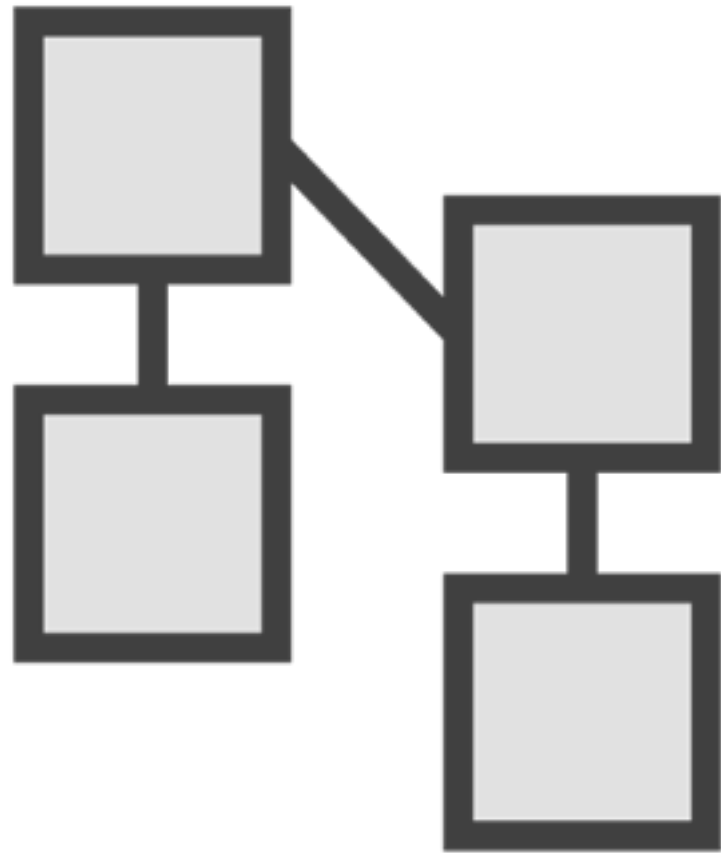
**Persistent Store**

- Add, remove, rename an Entity
- Add, remove, rename an Attribute
- Add, remove, rename a Relationship
- Other data model changes

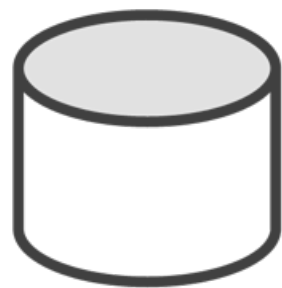
# Workflow



- 1. Create a new data model version**
- 2. Enable migrations**

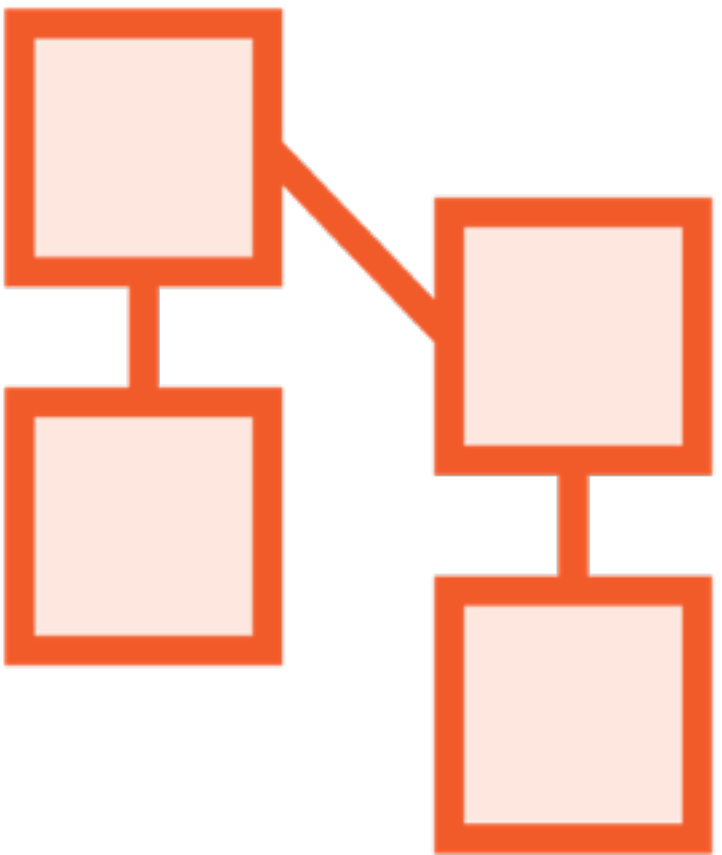


**Existing Data Model**

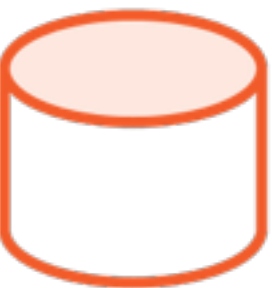


(Version 1)

New Model Version

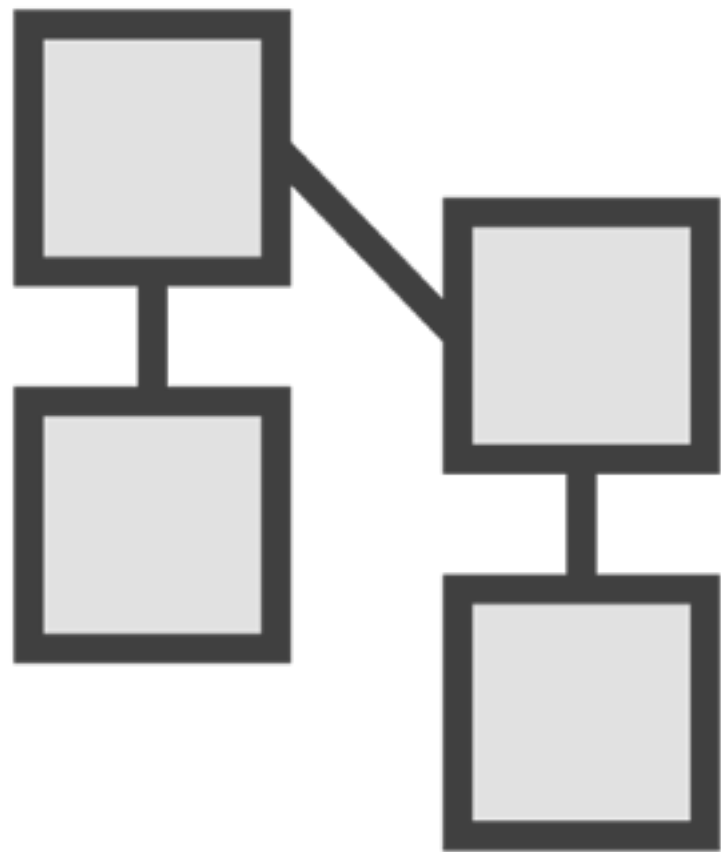


**Data Model w/ Changes**

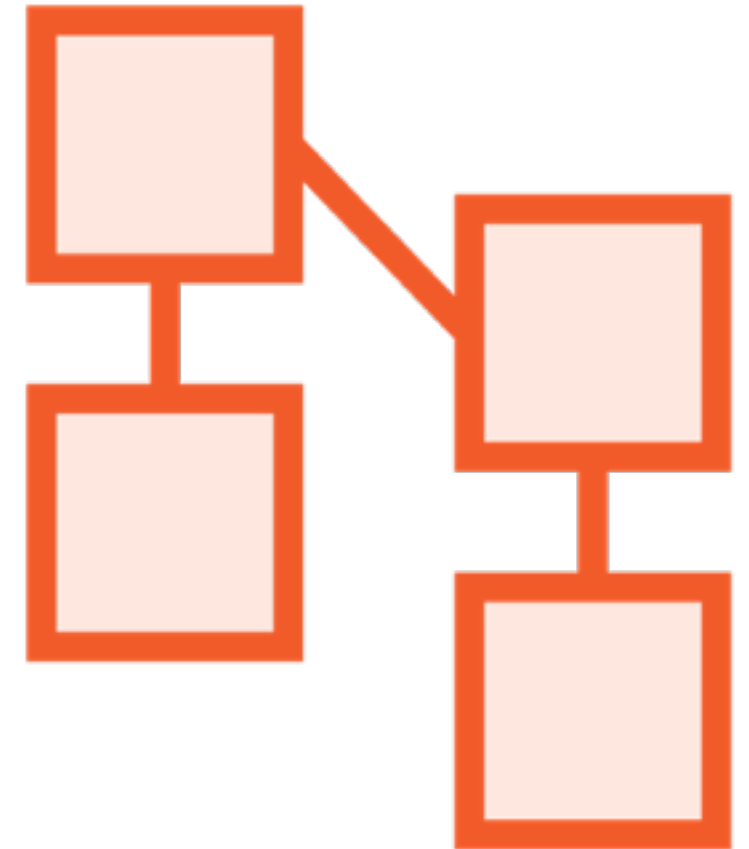
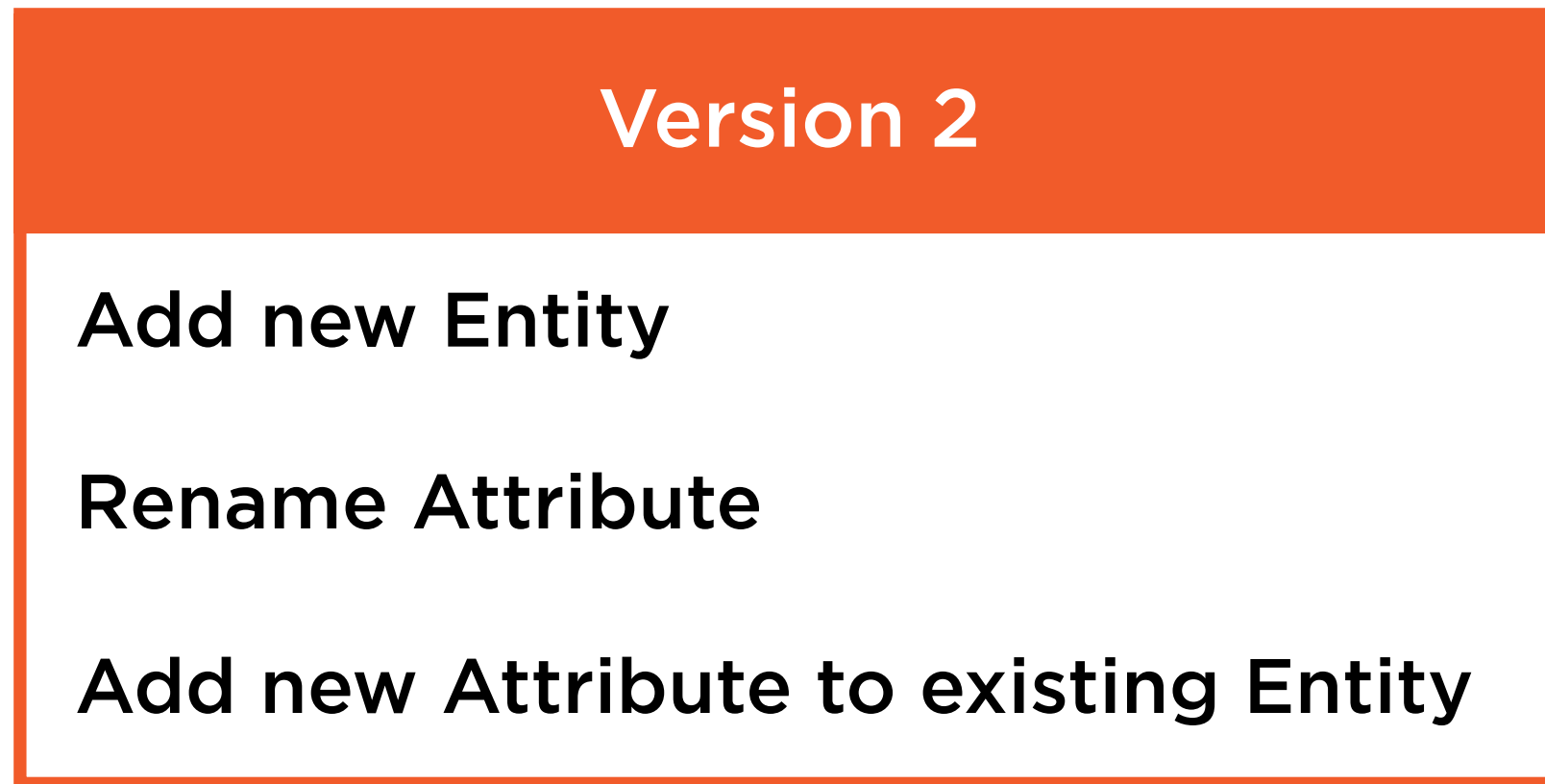
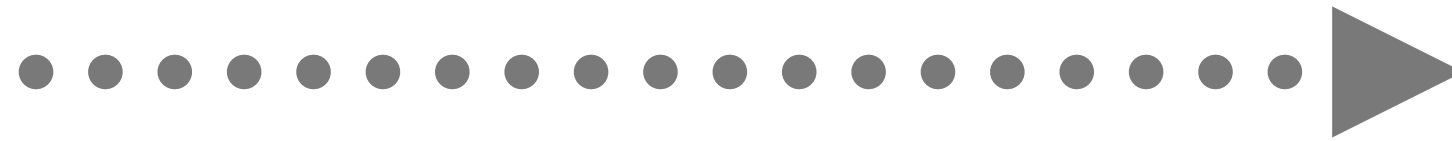


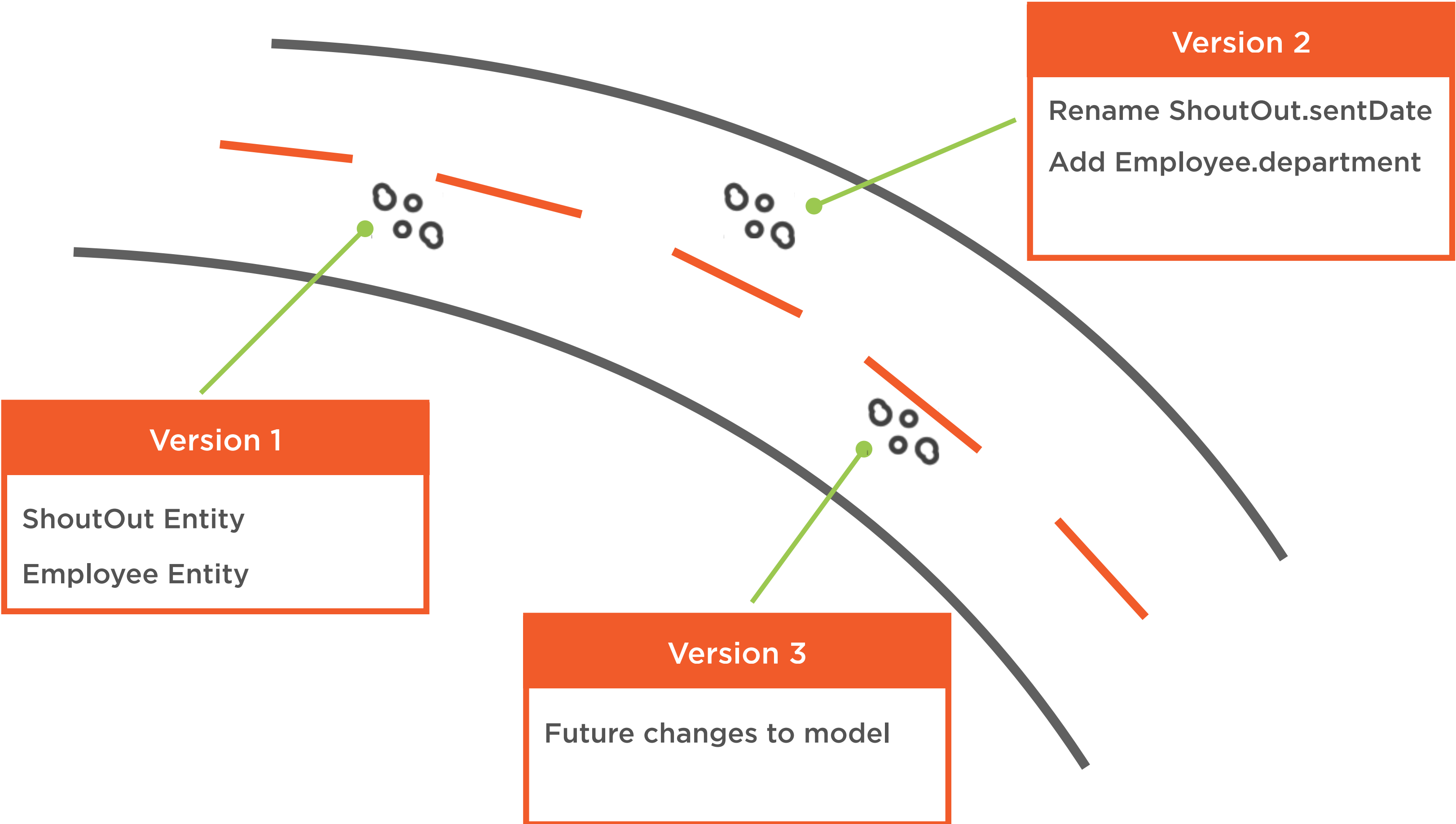
(Version 2)





(Version 1)

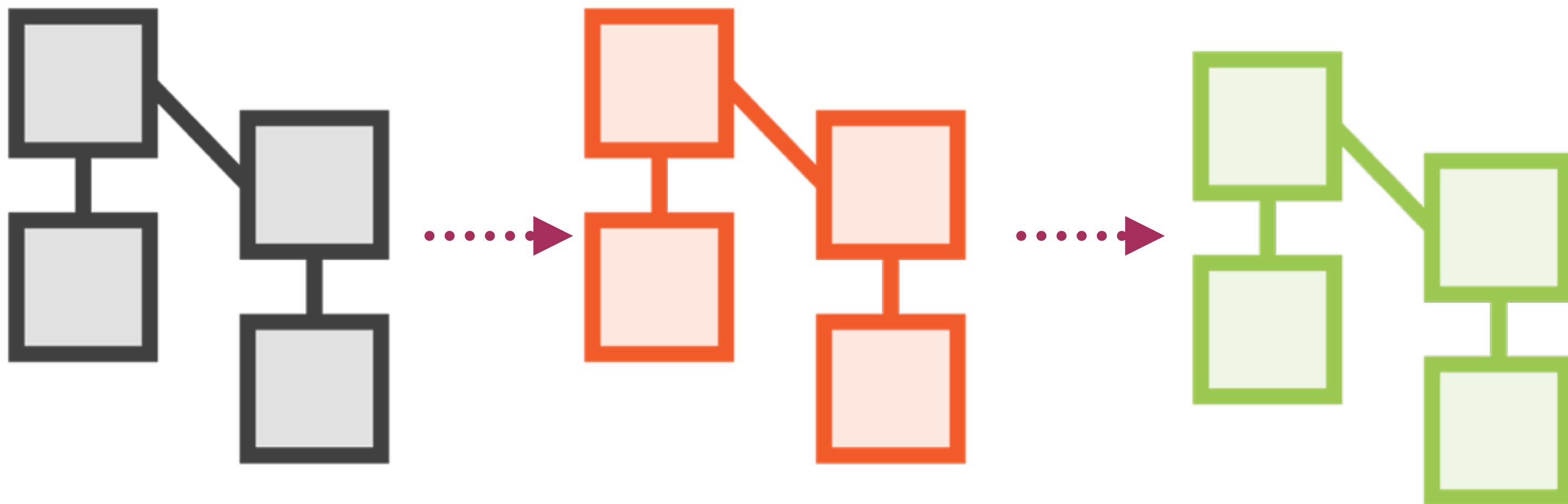




How does Core Data migrate  
between model versions?

# Migrating Data to New Data Model Versions

---



# Types of Migrations

**Lightweight Migration**

**Mapping Model**

# Types of Migrations

**Lightweight Migration**



# Lightweight Migration Scenarios



1. Add, remove, or rename Entities, Attributes, and Relationships
2. Make non-optional Attribute *optional*
3. Make *optional* Attribute *non-optional* (with default value)
4. Add, remove, or rename Entities within a hierarchy
5. Move Attributes up or down in a hierarchy
6. <http://bit.ly/LightweightMigrationScenarios>



```
try! psc.addPersistentStore(  
    ofType: NSSQLiteStoreType,  
    configurationName: nil,  
    at: storeURL,  
    options: nil)
```

## Enabling Lightweight Migrations

Create an options dictionary to pass to the persistent store coordinator's `addPersistentStore` method

```
let pscOptions = [
]

try! psc.addPersistentStore(
    ofType: NSSQLiteStoreType,
    configurationName: nil,
    at: storeURL,
    options: pscOptions)
```

## Enabling Lightweight Migrations

Create an options dictionary to pass to the persistent store coordinator's `addPersistentStore` method

Add `NSMigratePersistentStoresAutomaticallyOption` and `NSInferMappingModelAutomaticallyOption` keys to the dictionary

```
let pscOptions = [
    NSMigratePersistentStoresAutomaticallyOption : true,
    NSInferMappingModelAutomaticallyOption : true
]

try! psc.addPersistentStore(
    ofType: NSSQLiteStoreType,
    configurationName: nil,
    at: storeURL,
    options: pscOptions)
```

## Enabling Lightweight Migrations

Create an options dictionary to pass to the persistent store coordinator's `addPersistentStore` method

Add `NSMigratePersistentStoresAutomaticallyOption` and `NSInferMappingModelAutomaticallyOption` keys to the dictionary

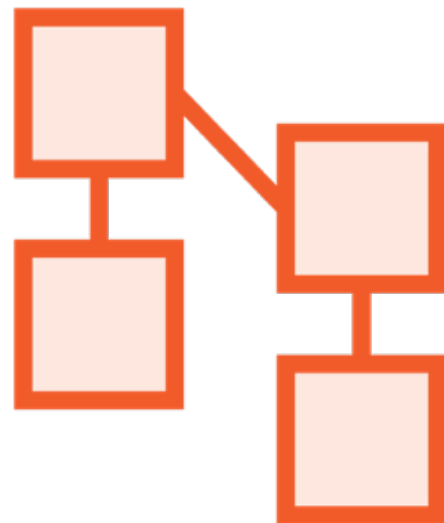
# Demo

**Create a new version of the data model**

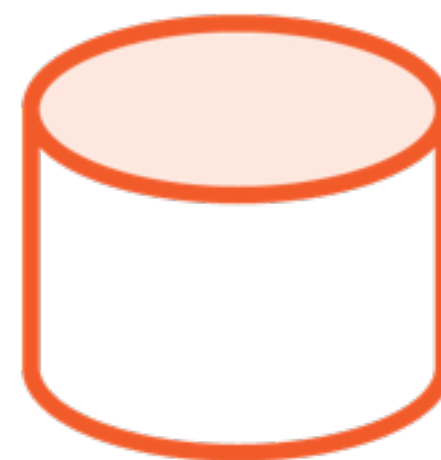
**Make changes to the data model**

- **Add Employee.department Attribute**
- **Rename ShoutOut.sentDate to ShoutOut.sentOn**

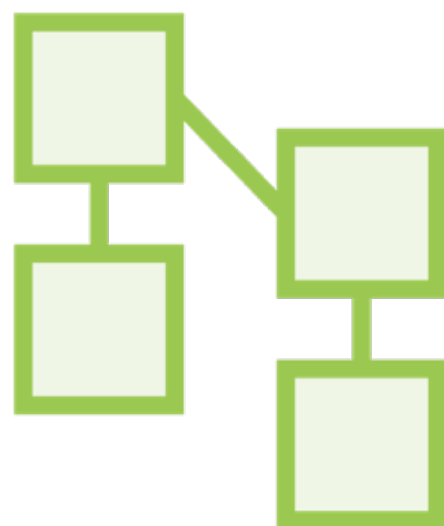
**Enable lightweight migrations in code**



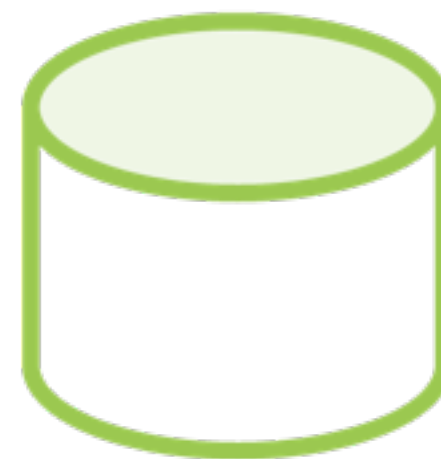
**Remove sentDate**



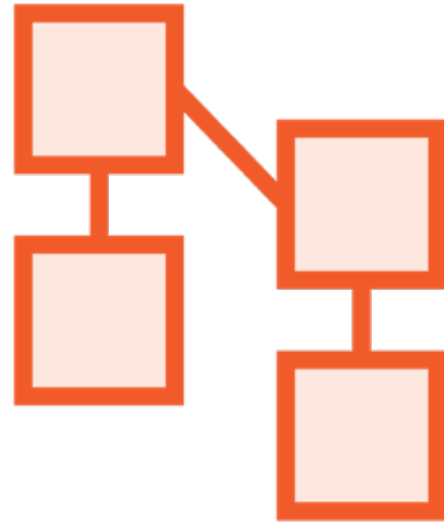
**Remove sentDate**



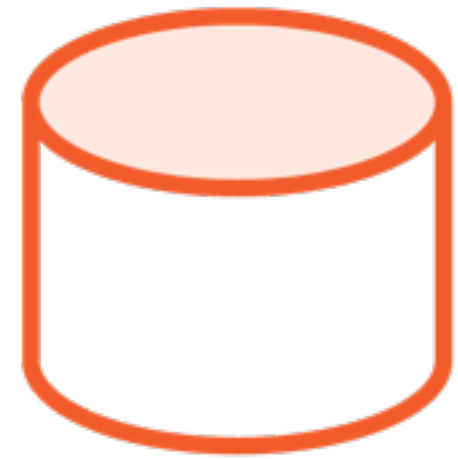
**Add sentOn**



**Add sentOn**



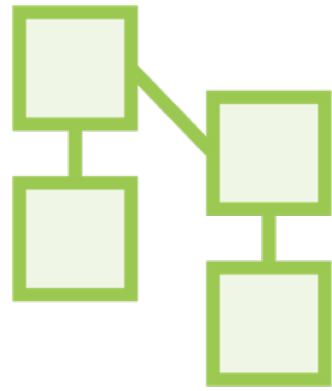
**Remove sentDate**



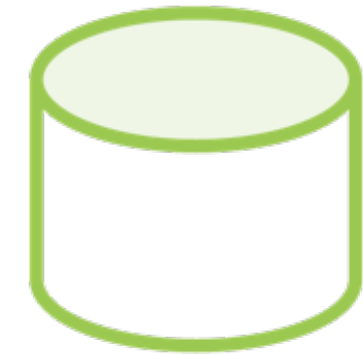
**Remove sentDate**



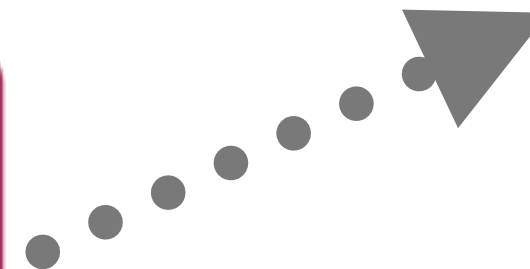
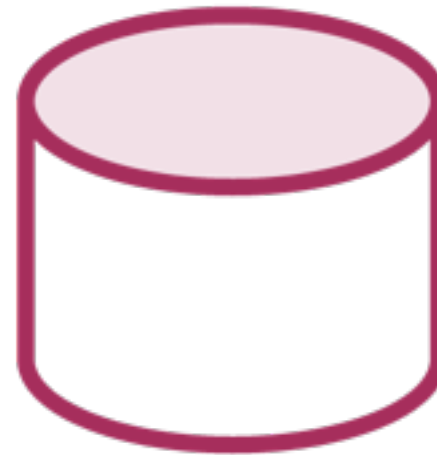
**Data stored in sentDate will be lost!**



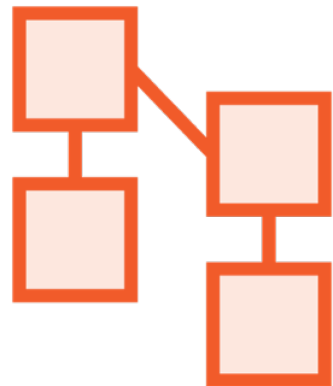
Add sentOn



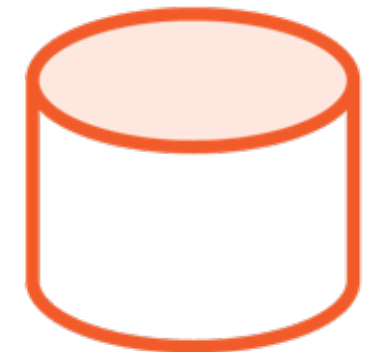
Add sentOn



Move data in sentDate to sentOn



Remove sentDate



Remove sentDate

# Demo

**Show and synchronize ShoutOuts  
grouped by department in table view**

**Implement remaining  
NSFetchedResultsControllerDelegate  
methods**



# Summary

**Responded to change requests by creating new data model version**

**Implemented lightweight migrations**

**Completed all major features of the ShoutOut app involving Core Data!**

**Coming up: A look to the future by examining Core Data changes with Apple's latest platforms**