In defense of Core Data

Core Dataを擁護する

Donny Wals

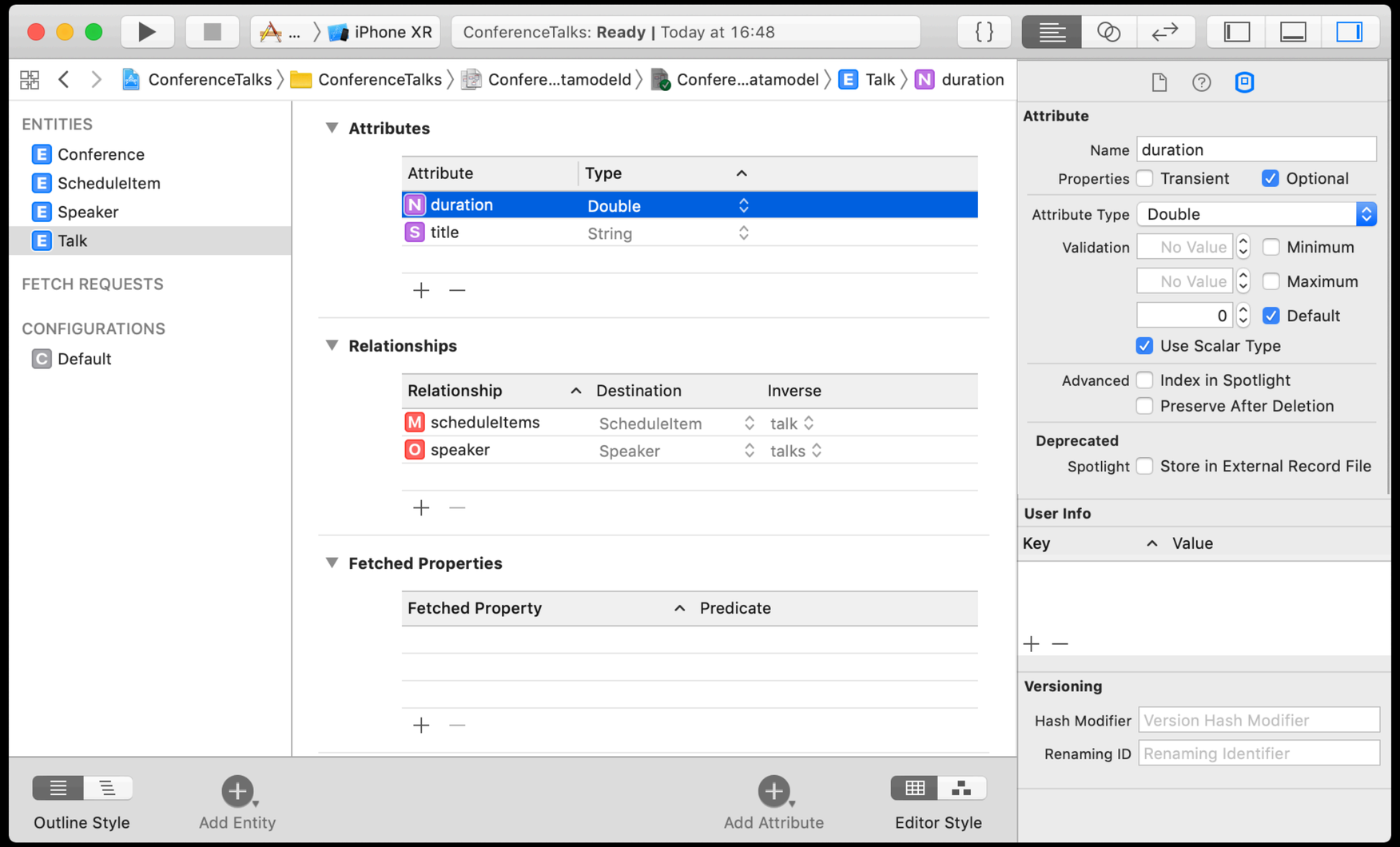
Agenda

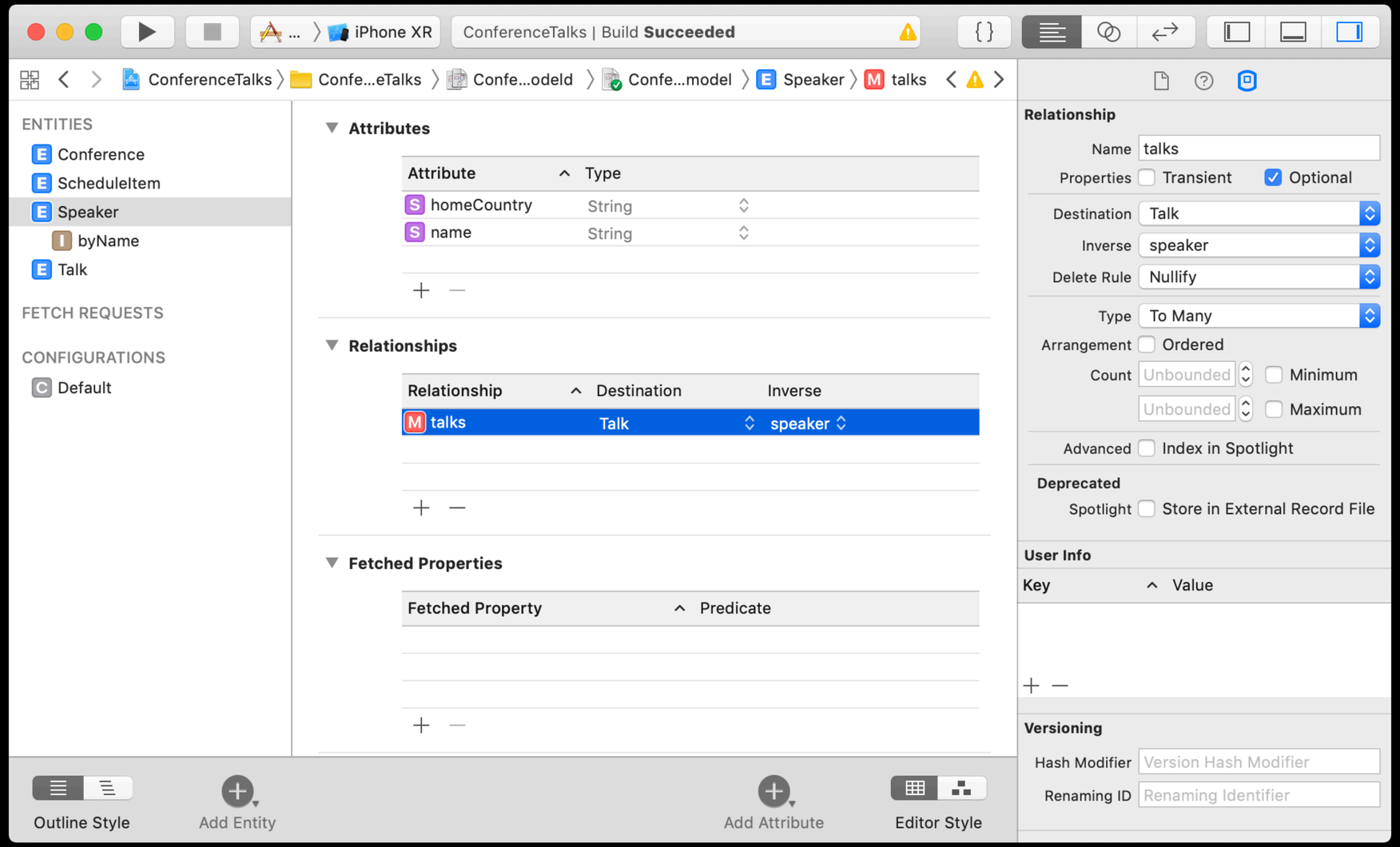
- Setting up a Core Data stack
  コアデータスタックの設定
- Updating your data model
  データモデルを更新する
- Using multiple managed object contexts
  複数の管理オブジェクトコンテキストを使用する
- Measuring and improving performance
  パフォーマンスの測定と改善

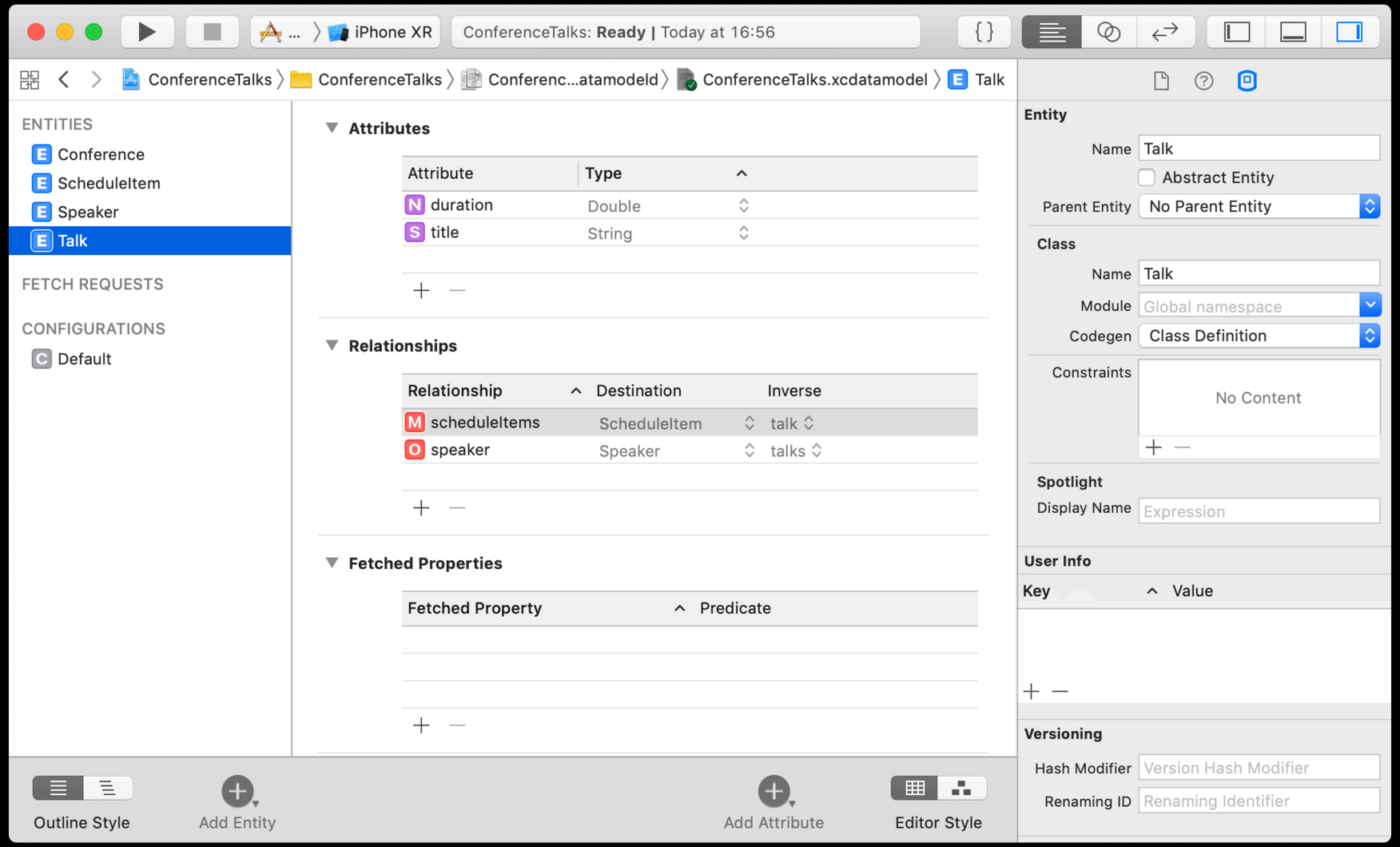
Setting up a Core Data stack

コアデータスタックの設定

```
lazy var persistentContainer: NSPersistentContainer = {
  let container = NSPersistentContainer(name: “ConferenceTalks")
  container.viewContext.automaticallyMergesChangesFromParent = true

  container.loadPersistentStores { storeDescription, error in
    if let error = error as NSError? {
      fatalError("Unresolved error \(error), \(error.userInfo)")
    }
  }

  return container
}()
```


iPhone XR
ConferenceTalks: Ready | Today at 16:48
ConferenceTalks
ConferenceTalks
Confere...tamodeld
Confere...atamodel
Talk
duration
ENTITIES
Conference
ScheduleItem
Speaker
Talk
FETCH REQUESTS
CONFIGURATIONS
Default
Attributes
Attribute
Type
duration
Double
title
String
Relationships
Relationship
Destination
Inverse
scheduleItems
ScheduleItem
talk
speaker
Speaker
talks
Fetched Properties
Fetched Property
Predicate
Attribute
Name
duration
Properties
Transient
Optional
Attribute Type
Double
Validation
No Value
Minimum
No Value
Maximum
0
Default
Use Scalar Type
Advanced
Index in Spotlight
Preserve After Deletion
Deprecated
Spotlight
Store in External Record File
User Info
Key
Value
Versioning
Hash Modifier
Version Hash Modifier
Renaming ID
Renaming Identifier
Outline Style
Add Entity
Add Attribute
Editor Style


iPhone XR
ConferenceTalks | Build Succeeded
ConferenceTalks
Confe...eTalks
Confe...odeld
Confe...model
Speaker
talks
ENTITIES
Conference
ScheduleItem
Speaker
byName
Talk
FETCH REQUESTS
CONFIGURATIONS
Default
Attributes
Attribute
Type
homeCountry
String
name
String
Relationships
Relationship
Destination
Inverse
talks
Talk
speaker
Fetched Properties
Fetched Property
Predicate
Relationship
Name
talks
Properties
Transient
Optional
Destination
Talk
Inverse
speaker
Delete Rule
Nullify
Type
To Many
Arrangement
Ordered
Count
Unbounded
Minimum
Unbounded
Maximum
Advanced
Index in Spotlight
Deprecated
Spotlight
Store in External Record File
User Info
Key
Value
Versioning
Hash Modifier
Version Hash Modifier
Renaming ID
Renaming Identifier
Outline Style
Add Entity
Add Attribute
Editor Style


iPhone XR
ConferenceTalks: Ready | Today at 16:56
ConferenceTalks
ConferenceTalks
Conferenc...atamodeld
ConferenceTalks.xcdatamodel
Talk
ENTITIES
Conference
ScheduleItem
Speaker
Talk
FETCH REQUESTS
CONFIGURATIONS
Default
Attributes
Attribute
Type
duration
Double
title
String
Relationships
Relationship
Destination
Inverse
scheduleItems
ScheduleItem
talk
speaker
Speaker
talks
Fetched Properties
Fetched Property
Predicate
Entity
Name
Talk
Abstract Entity
Parent Entity
No Parent Entity
Class
Name
Talk
Module
Global namespace
Codegen
Class Definition
Constraints
No Content
Spotlight
Display Name
Expression
User Info
Key
Value
Versioning
Hash Modifier
Version Hash Modifier
Renaming ID
Renaming Identifier
Outline Style
Add Entity
Add Attribute
Editor Style

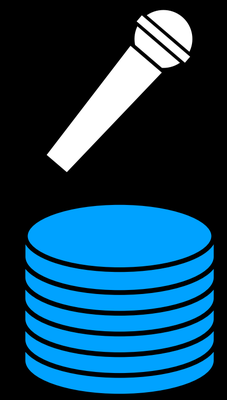
Let’s turn this into an app! ✌🏻

これをアプリに変えましょう！ ✌🏻

Using a fetched results controller

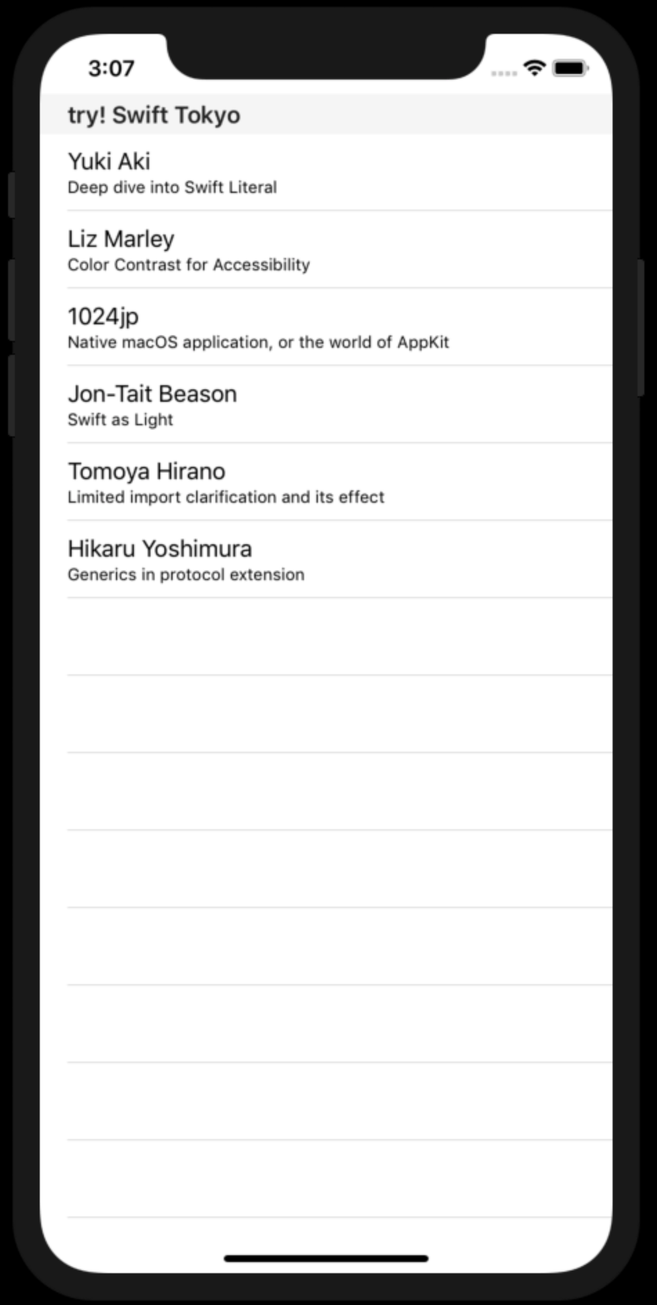
fetched results controller を使う

```
let req: NSFetchRequest<ScheduleItem> = ScheduleItem.fetchRequest()
req.sortDescriptors = [
  NSSortDescriptor(key: "conference.name", ascending: true),
  NSSortDescriptor(key: "start", ascending: true)]

self.fetchedResultsController =
  NSFetchedResultsController(fetchRequest: req,
                             managedObjectContext:
                               persistentContainer.viewContext,
                             sectionNameKeyPath: "conference.name",
                             cacheName: nil)

fetchedResultsController.delegate = self

do {
  try fetchedResultsController.performFetch()
  tableView.reloadData()
} catch { /* handle errors */ }
```



Underlying data store

Fetched results controller



```swift
func controllerWillChangeContent(_ controller: NSFetchedResultsController<NSFetchRequestResult>) {
  tableView.beginUpdates()
}


func controllerDidChangeContent(_ controller: NSFetchedResultsController<NSFetchRequestResult>) {
  tableView.endUpdates()
}


func controller(_ controller: NSFetchedResultsController<NSFetchRequestResult>,
                didChange anObject: Any, at indexPath: IndexPath?,
                for type: NSFetchedResultsChangeType, newIndexPath: IndexPath?) {
  switch type {
  case .insert:
    guard let insertIndex = newIndexPath else { return }
    tableView.insertRows(at: [insertIndex], with: .automatic)
  case .delete:
    guard let deleteIndex = indexPath else { return }
    tableView.deleteRows(at: [deleteIndex], with: .automatic)
  case .move:
    guard let fromIndex = indexPath, let toIndex = newIndexPath else { return }
    tableView.moveRow(at: fromIndex, to: toIndex)
  case .update:
    guard let updateIndex = indexPath else { return }
    tableView.reloadRows(at: [updateIndex], with: .automatic)
  }
}
```

Awesome! 👍🏼

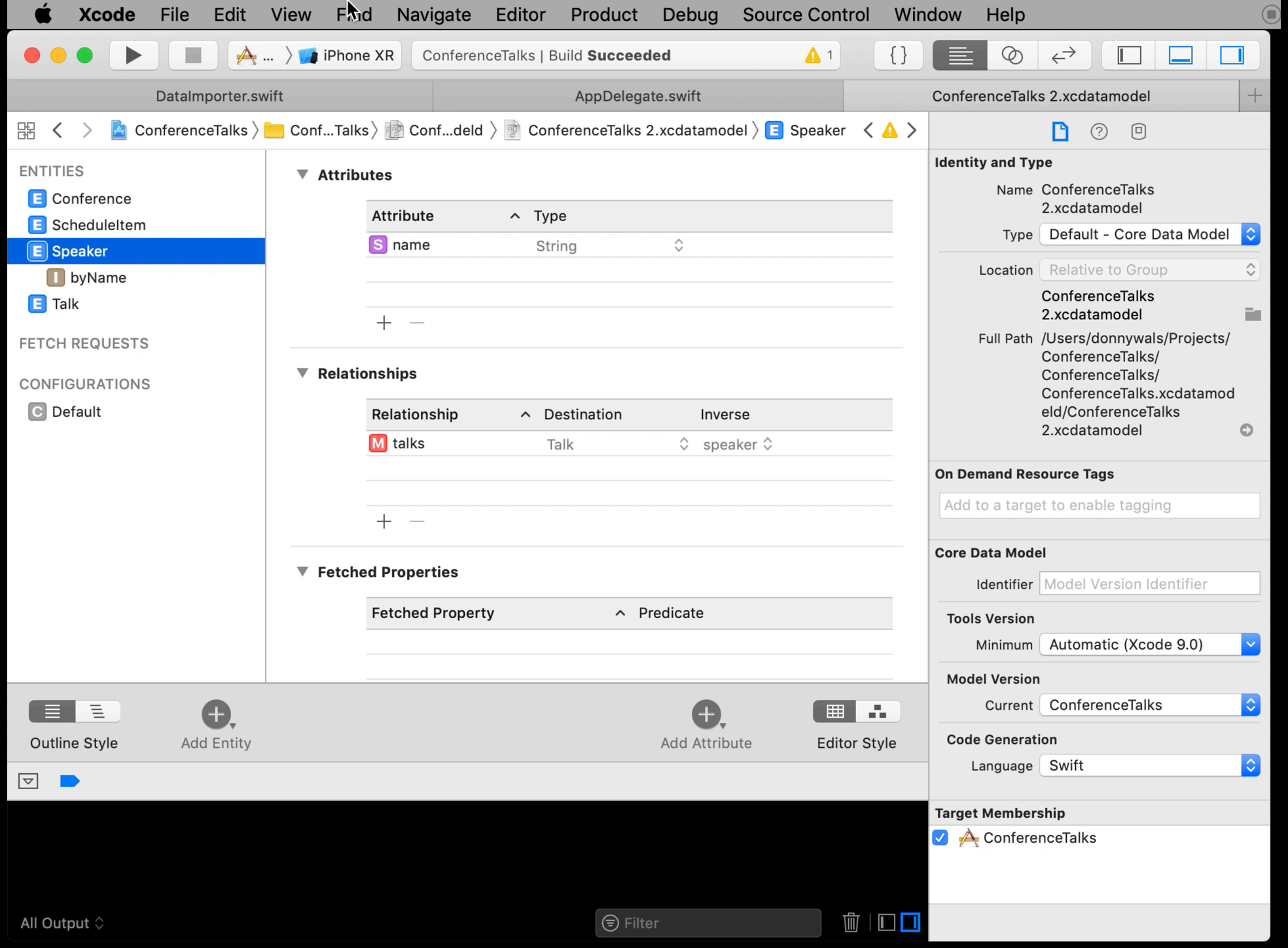
驚くばかり！ 👍🏼

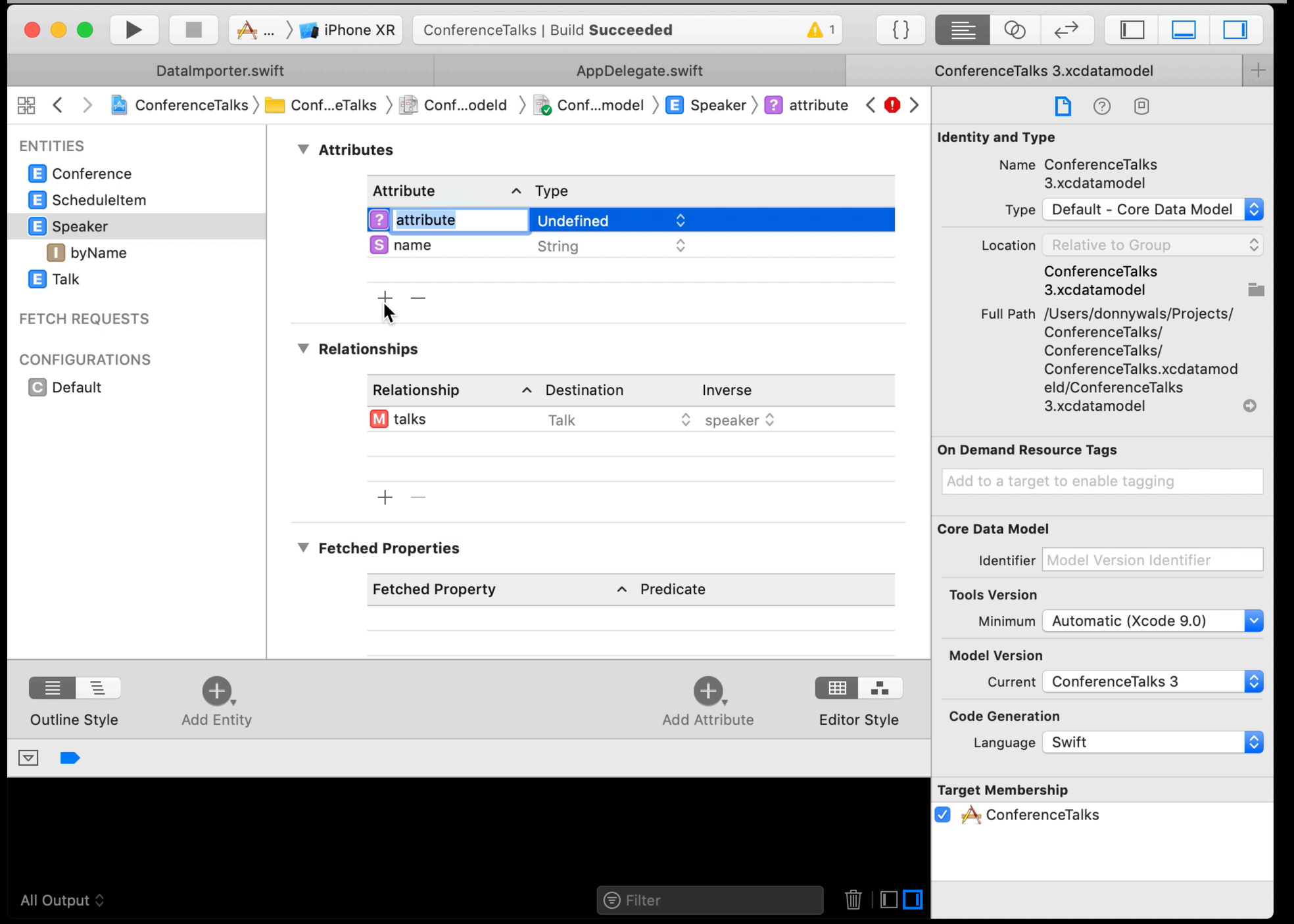
But what if I update my model?

しかし、モデルを更新したらどうなるでしょうか。

I'll have to do a migration right?

移行を実行する必要がありますか。


Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help
iPhone XR
ConferenceTalks | Build Succeeded
DataImporter.swift
AppDelegate.swift
ConferenceTalks 2.xcdatamodel
ConferenceTalks
Conf...Talks
Conf...deld
ConferenceTalks 2.xcdatamodel
Speaker
ENTITIES
Conference
ScheduleItem
Speaker
byName
Talk
FETCH REQUESTS
CONFIGURATIONS
Default
Attributes
Attribute
Type
name
String
Relationships
Relationship
Destination
Inverse
talks
Talk
speaker
Fetched Properties
Fetched Property
Predicate
Outline Style
Add Entity
Add Attribute
Editor Style
All Output
Filter
Identity and Type
Name ConferenceTalks 2.xcdatamodel
Type Default - Core Data Model
Location Relative to Group
ConferenceTalks 2.xcdatamodel
Full Path /Users/donnywals/Projects/ConferenceTalks/ConferenceTalks/ConferenceTalks.xcdatamodeld/ConferenceTalks 2.xcdatamodel
On Demand Resource Tags
Add to a target to enable tagging
Core Data Model
Identifier Model Version Identifier
Tools Version
Minimum Automatic (Xcode 9.0)
Model Version
Current ConferenceTalks
Code Generation
Language Swift
Target Membership
ConferenceTalks


Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help
iPhone XR
ConferenceTalks | Build Succeeded
DataImporter.swift
AppDelegate.swift
ConferenceTalks 3.xcdatamodel
ConferenceTalks
Conf...eTalks
Conf...odeld
Conf...model
Speaker
attribute
ENTITIES
Conference
ScheduleItem
Speaker
byName
Talk
FETCH REQUESTS
CONFIGURATIONS
Default
Attributes
Attribute
Type
attribute
Undefined
name
String
Relationships
Relationship
Destination
Inverse
talks
Talk
speaker
Fetched Properties
Fetched Property
Predicate
Outline Style
Add Entity
Add Attribute
Editor Style
Identity and Type
Name
ConferenceTalks 3.xcdatamodel
Type
Default - Core Data Model
Location
Relative to Group
ConferenceTalks 3.xcdatamodel
Full Path
/Users/donnywals/Projects/ConferenceTalks/ConferenceTalks/ConferenceTalks.xcdatamodeld/ConferenceTalks 3.xcdatamodel
On Demand Resource Tags
Add to a target to enable tagging
Core Data Model
Identifier
Model Version Identifier
Tools Version
Minimum
Automatic (Xcode 9.0)
Model Version
Current
ConferenceTalks 3
Code Generation
Language
Swift
Target Membership
ConferenceTalks
All Output
Filter

Rules for automatic migrations

自動移行のルール

- Works for simple changes only

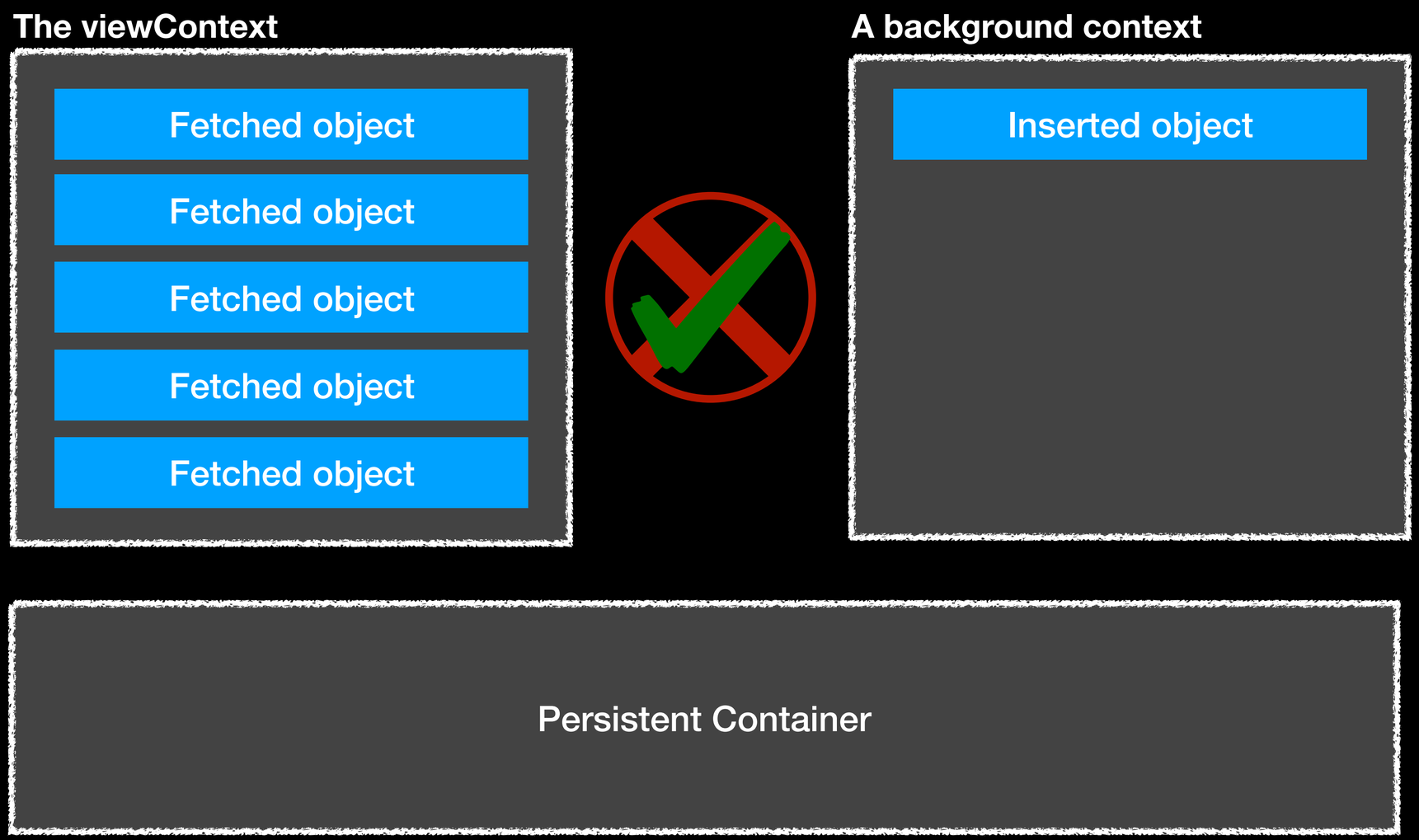
  単純な変更に対してのみ機能します

- Always test your update to be absolutely sure everything is working as expected

  常にすべてが期待通りに機能していることを確実にするためにアップデートをテストしてください

A multi-context setup

マルチコンテキスト設定



Using a background context

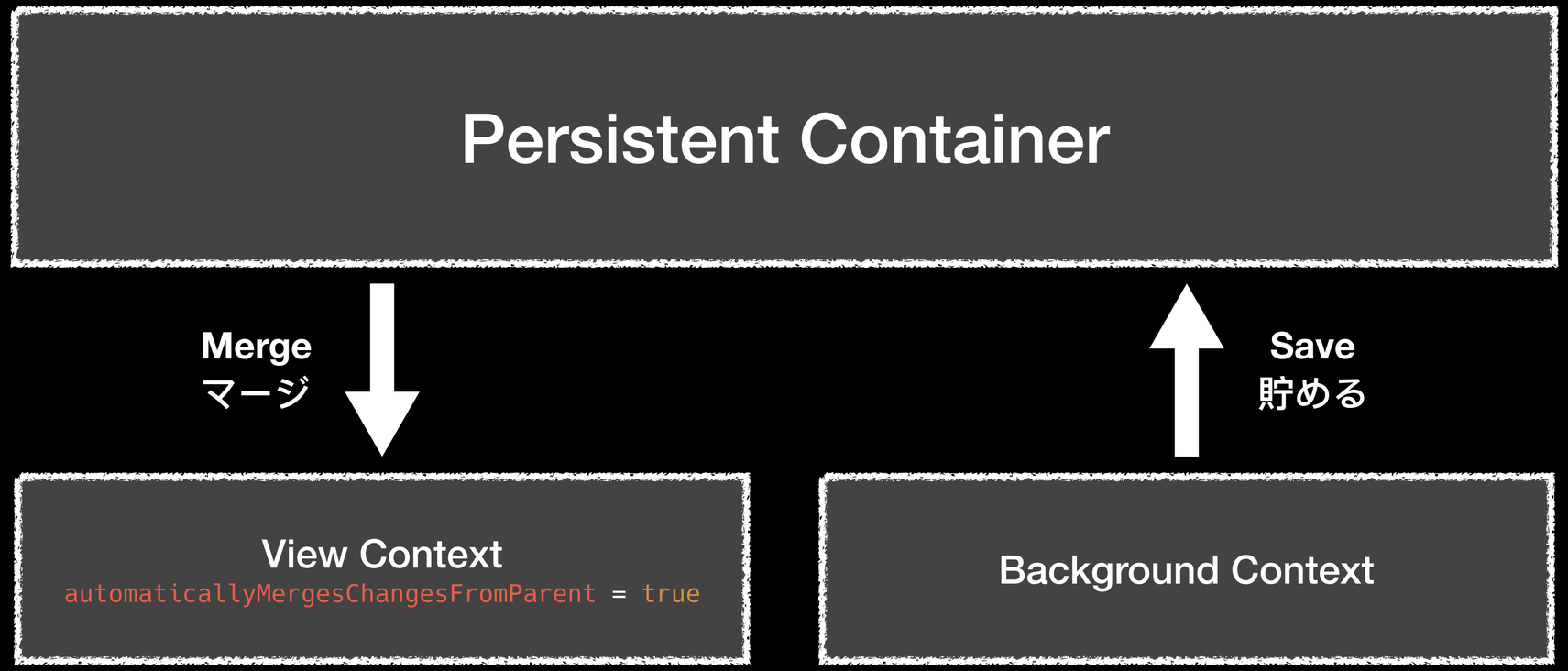
バックグラウンドコンテキストを使用する

```
let backgroundContext = persistentContainer.newBackgroundContext()
backgroundContext.perform {
  // perform background work asynchronously here
}
```

or

または

```
persistentContainer.performBackgroundTask { context in
  // perform background work asynchronously here
}
```

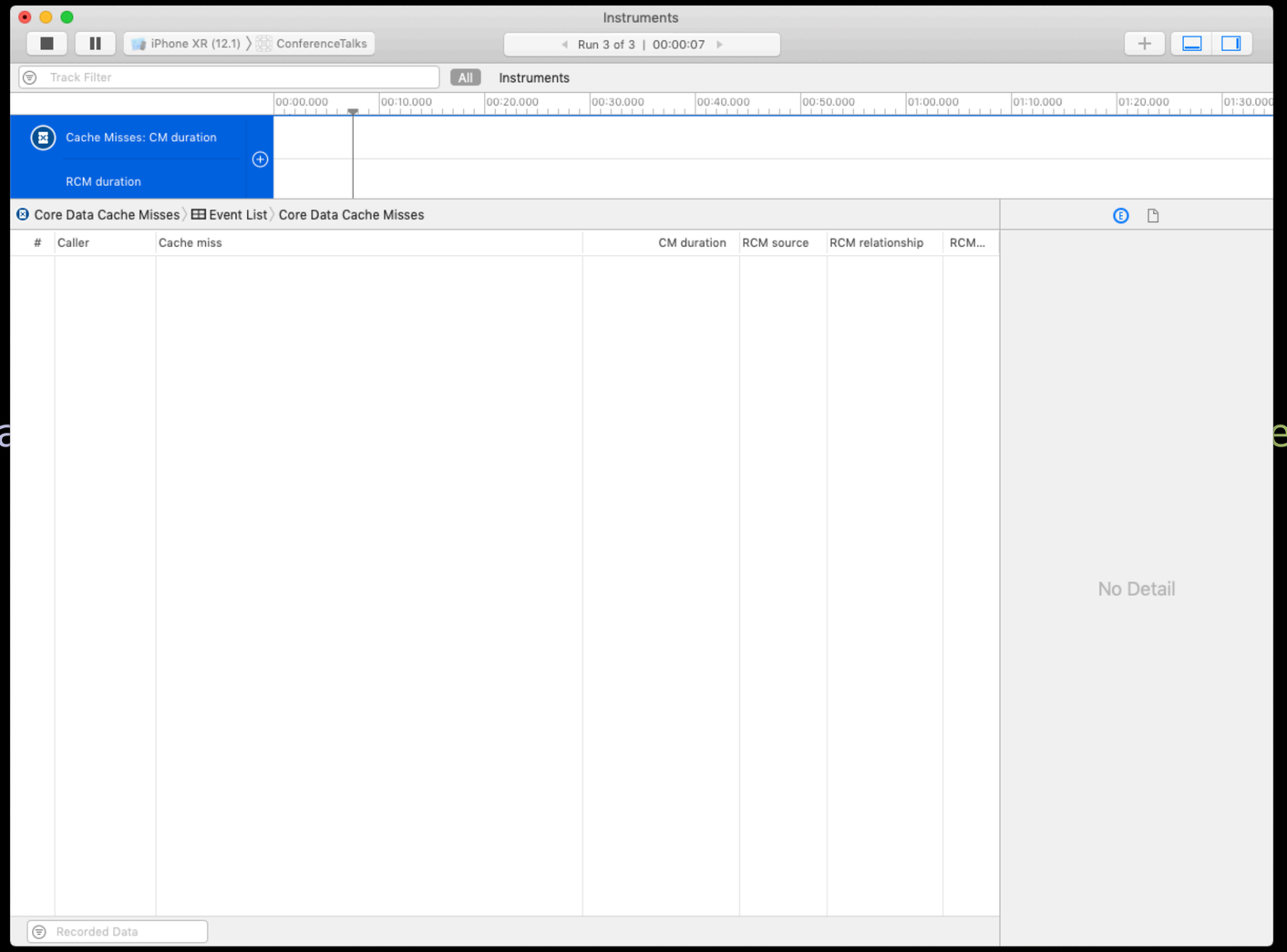

Persistent Container
Merge
マージ
Save
貯める
View Context
automaticallyMergesChangesFromParent = true
Background Context

Performance ⏱

パフォーマンス ⏱

Optimizing your fetch requests

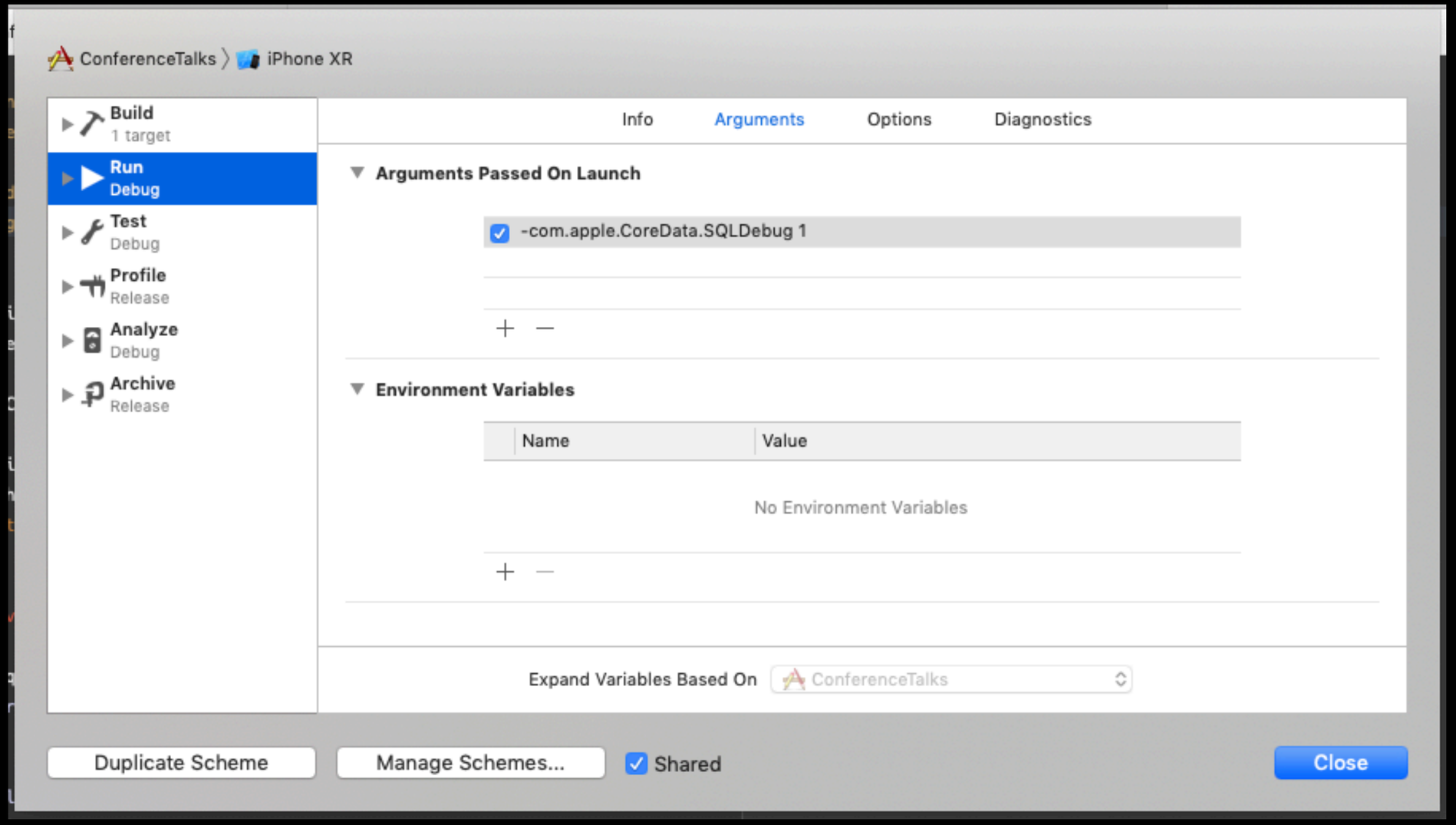
フェッチ要求を最適化する




```
req.rela                                                    er"]
```

Analyzing your queries

クエリを分析する



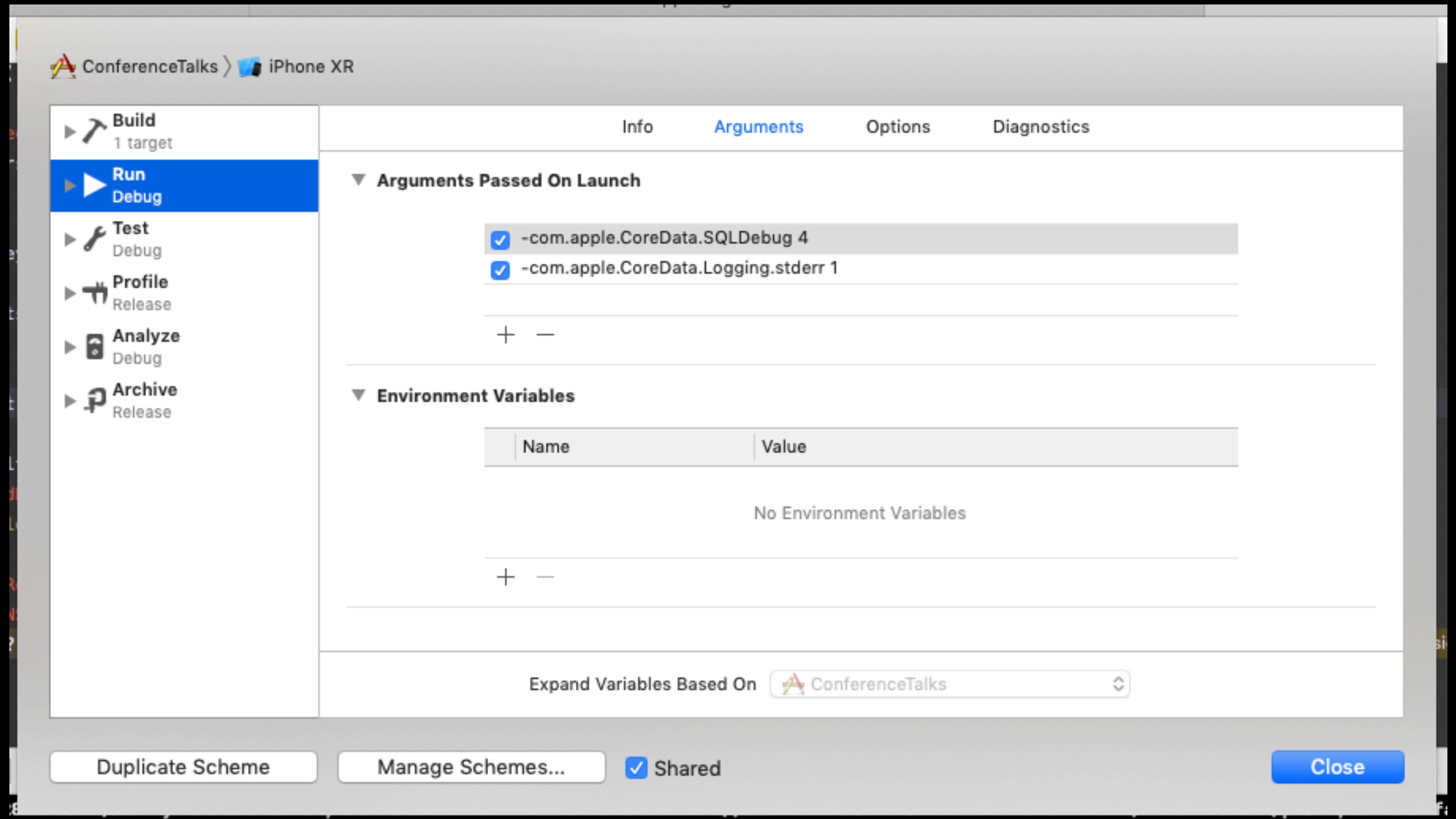
Analyzing your queries

クエリを分析する

```
CoreData: warning:  View context accessed for persistent container ConferenceTalks with no stores loaded
CoreData: annotation: Connecting to sqlite database file at "/Users/donnywals/Library/Developer/CoreSimulator/Devices/9F01D8CD-1203-4FA4-A625-8394108A3B16/data/Containers/Data/Application/1096D89D-6A1B-43CF-AC00-92DAA6E25D79/Library/Application Support/ConferenceTalks.sqlite"
CoreData: sql: SELECT TBL_NAME FROM SQLITE_MASTER WHERE TBL_NAME = 'Z_METADATA'
CoreData: sql: pragma recursive_triggers=1
CoreData: sql: pragma journal_mode=wal
CoreData: sql: SELECT Z_VERSION, Z_UUID, Z_PLIST FROM Z_METADATA
CoreData: sql: SELECT TBL_NAME FROM SQLITE_MASTER WHERE TBL_NAME = 'Z_METADATA'
CoreData: sql: SELECT TBL_NAME FROM SQLITE_MASTER WHERE TBL_NAME = 'Z_MODELCACHE'
CoreData: sql: SELECT TBL_NAME FROM SQLITE_MASTER WHERE TBL_NAME = 'ACHANGE'
CoreData: sql: SELECT 0, t0.Z_PK, t0.Z_OPT, t0.ZEND, t0.ZSTART, t0.ZCONFERENCE, t0.ZTALK, t0.Z_FOK_CONFERENCE FROM ZSCHEDULEITEM t0 LEFT OUTER JOIN ZCONFERENCE t1 ON t0.ZCONFERENCE = t1.Z_PK ORDER BY t1.ZNAME, t0.ZSTART
CoreData: annotation: sql connection fetch time: 0.0003s
CoreData: annotation: Bound intarray _Z_intarray0
CoreData: annotation: Bound intarray values.
CoreData: sql: SELECT 0, t0.Z_PK, t0.Z_OPT, t0.ZTITLE, t0.ZSPEAKER FROM ZTALK t0 WHERE  t0.Z_PK IN (SELECT * FROM _Z_intarray0)
CoreData: annotation: sql connection fetch time: 0.0006s
CoreData: annotation: total fetch execution time: 0.0008s for 15 rows.
CoreData: annotation: Prefetching with key 'talk'.  Got 15 rows.
CoreData: annotation: Bound intarray values.
CoreData: sql: SELECT 0, t0.Z_PK, t0.Z_OPT, t0.ZNAME FROM ZSPEAKER t0 WHERE  t0.Z_PK IN (SELECT * FROM _Z_intarray0)
CoreData: annotation: sql connection fetch time: 0.0002s
CoreData: annotation: total fetch execution time: 0.0003s for 15 rows.
CoreData: annotation: Prefetching with key 'speaker'.  Got 15 rows.
CoreData: annotation: total fetch execution time: 0.0045s for 15 rows.
CoreData: sql: SELECT  t1.ZNAME, COUNT (DISTINCT  t0.Z_PK) FROM ZSCHEDULEITEM t0 LEFT OUTER JOIN ZCONFERENCE t1 ON t0.ZCONFERENCE = t1.Z_PK GROUP BY  t1.ZNAME ORDER BY t1.ZNAME
CoreData: annotation: sql connection fetch time: 0.0002s
CoreData: annotation: total fetch execution time: 0.0003s for 1 rows.
```

Analyzing your queries

クエリを分析する



In conclusion 🤩

結論としては 🤩