

Core Data Fundamentals with Swift

GETTING STARTED WITH CORE DATA



Andrew Bancroft

@andrewcbancroft www.andrewcbancroft.com

What should I use?

Where do I get
started?

Is Core Data the
right tool?

Why Core Data?

Convenient and Robust

Handles both small and large data requirements

Integrated

Developed by Apple and integrated into Xcode

Tenure

Boasts 10+ years of continual enhancement

Available

Use on all of Apple's platforms

Current

Staying up-to-date with the latest features of Swift

What Is Core Data?

Defining Core Data



Core Data is **not a database**



Core Data is **not an
Object-Relational Mapper (ORM)**

Definition

Core Data is a framework used to manage the model layer objects in your application.

It provides generalized and automated solutions to common tasks associated with object life cycle and object graph management, including persistence.

Definition

Core Data is a **framework** used to manage the model layer objects in your application.

It provides generalized and automated solutions to common tasks associated with object life cycle and object graph management, including persistence.

Definition

Core Data is a **framework** used to manage the model layer objects in your application.

It provides generalized and automated solutions to common tasks associated with **object life cycle** and object graph management, including persistence.

Definition

Core Data is a **framework** used to manage the model layer objects in your application.

It provides generalized and automated solutions to common tasks associated with **object life cycle** and **object graph management**, including persistence.

Definition

Core Data is a **framework** used to manage the model layer objects in your application.

It provides generalized and automated solutions to common tasks associated with **object life cycle** and **object graph management**, including **persistence**.

Defining Core Data

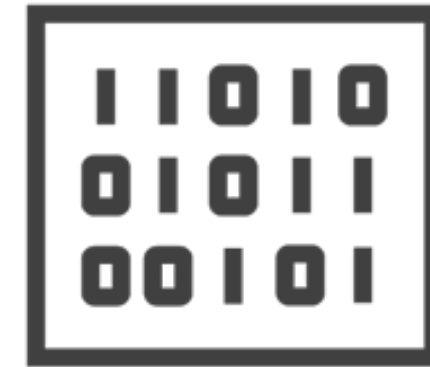


Framework

Core Data Is a Framework



NSManagedObjectContext



NSManagedObject



NSFetchedRequest



NSFetchedResultsController

Defining Core Data



Framework

Defining Core Data



Framework



Object Lifecycle

Core Data Manages Object Lifecycle



Create Objects

Change Objects

Delete Objects

Defining Core Data



Framework



Object Lifecycle

Defining Core Data



Framework

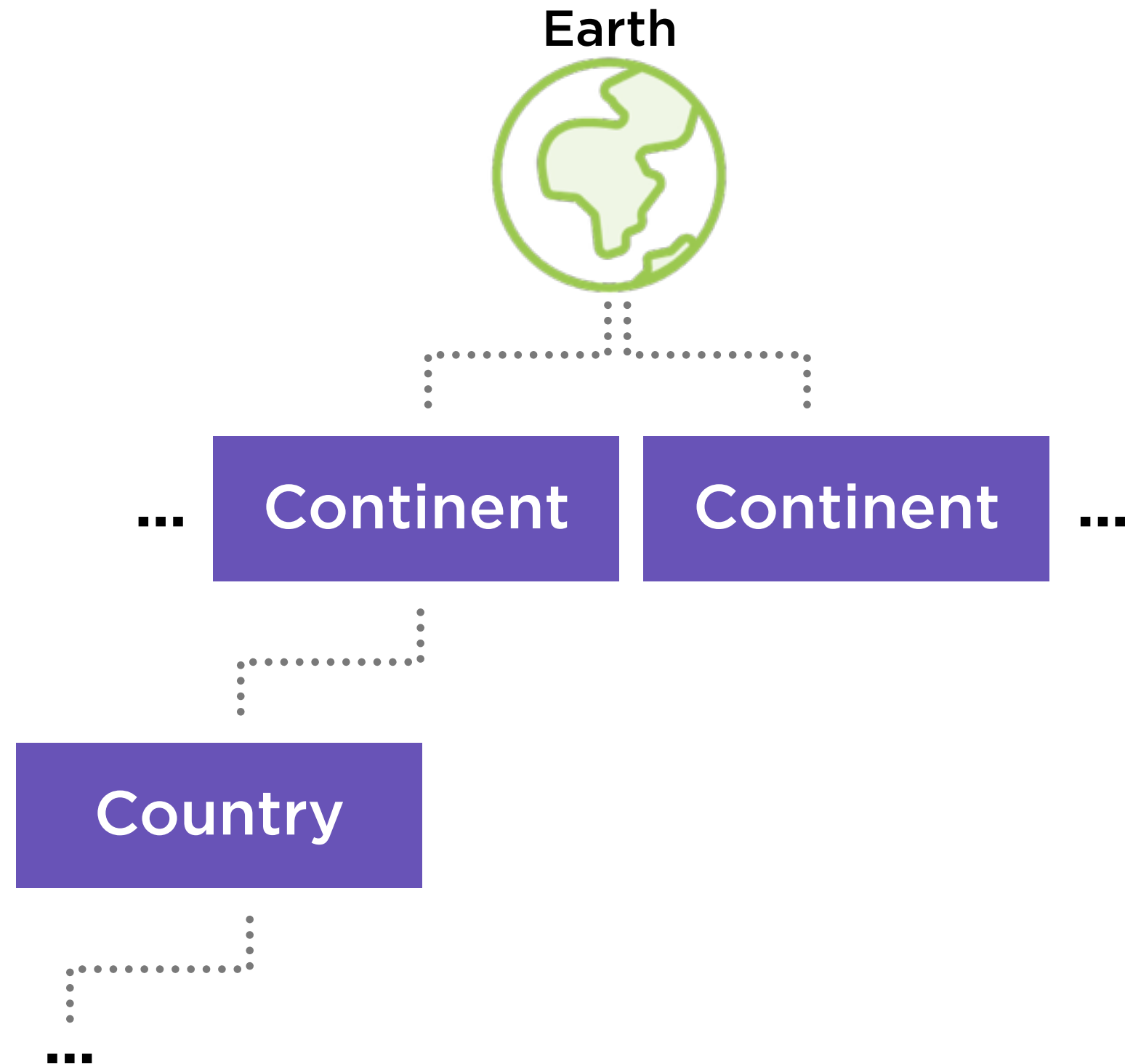


Object Lifecycle



Object Graphs

Core Data Models and Manages Object Graphs



Defining Core Data



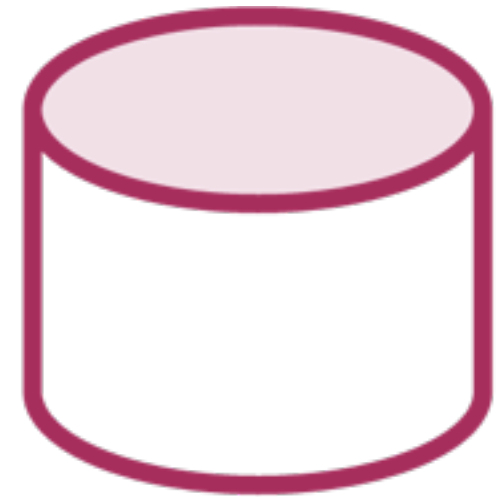
Framework



Object Lifecycle

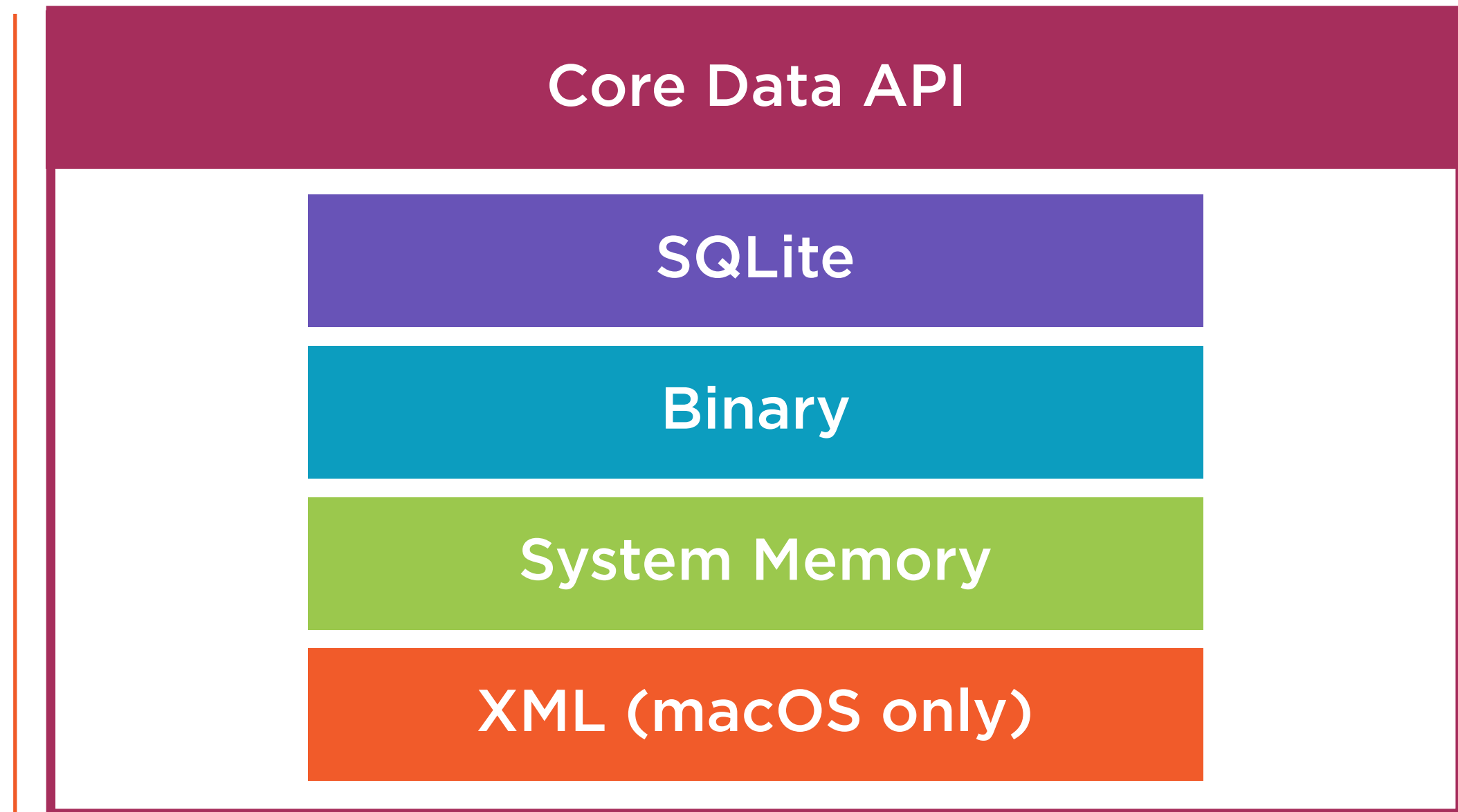
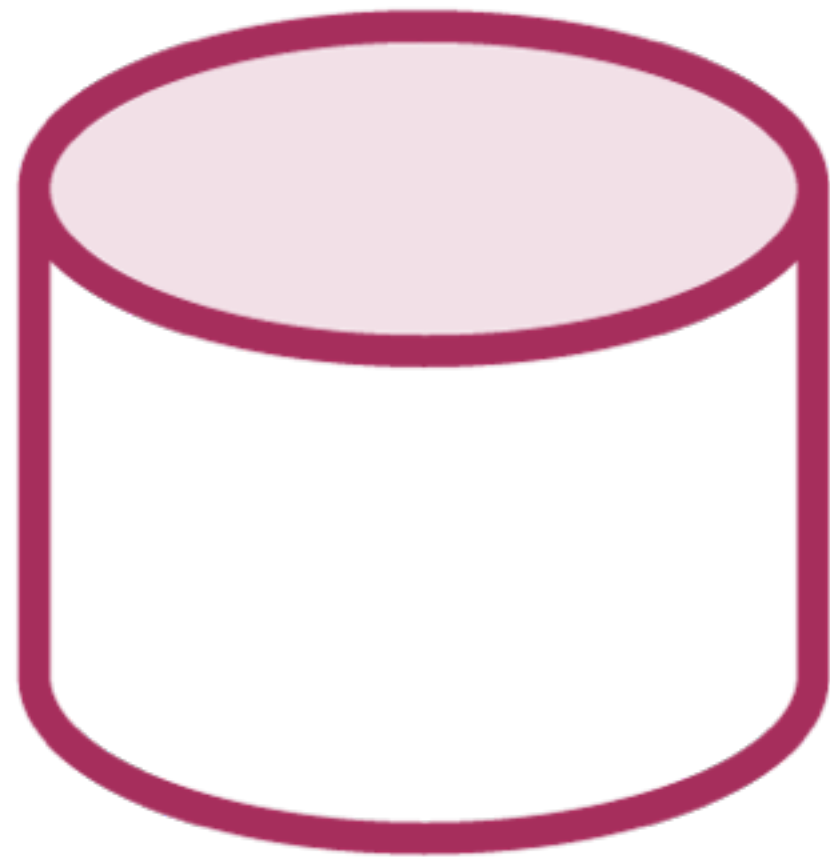


Object Graphs



Persistence

Core Data Abstracts Data Persistence



Is Core Data Overkill?

Is Core Data Overkill?

NSCoder

**Property lists
(plist)**

NSUserDefaults

Is Core Data Overkill?

Complexity of Data Model

**Future Data Management
Needs**

Type Creation

Course Prerequisites



Basic iOS development

Storyboards

Swift syntax

Demo

**Human Resources implementing new
Employee Recognition Program**

New app idea: ShoutOut

**Apple Enterprise Developer Program
members**

End product: ShoutOut app prototype

Demo

Tour ShoutOut Xcode starter project

**Storyboard with segues, IBActions, and
IBOutlets connected**

Course Goals

Set up the Core Data stack

Design a Core Data model in Xcode

Save, update, delete, and query data

Build and synchronize a user interface

Change and migrate data models

**Look to the future - Core Data on
Apple's latest platforms**