Introducing ClassKit
Session 215

Pavel Dudrenov, ClassKit team
Marin Eubanks, ClassKit team
General Overview

Framework Classes

Adopting ClassKit
General Overview

Framework Classes

Adopting ClassKit
General Overview
ClassKit Framework
Features
ClassKit Framework

Features

Focused on education
ClassKit Framework

Features

Focused on education

Define assignable content in your app
ClassKit Framework

Features

Focused on education

Define assignable content in your app

Record and report student progress
ClassKit Framework
Features

Focused on education
Define assignable content in your app
Record and report student progress
Respects student privacy
ClassKit Framework

Reasons to adopt
ClassKit Framework

Reasons to adopt

Provide better teacher workflow
ClassKit Framework

Reasons to adopt

Provide better teacher workflow

Provide insight for teachers
ClassKit Framework
Reasons to adopt

Provide better teacher workflow
Provide insight for teachers
Enables personalized instruction
ClassKit Framework
Reasons to adopt

Provide better teacher workflow
Provide insight for teachers
Enables personalized instruction
Competitive advantage
Apple School Manager
Admin portal

Managed Apple IDs
23,223 Managed Apple IDs

Domain: townshipschools.k12.edu

Student Format: S 1234567890 @townshipschools.k12.edu
5123456789@townshipschools.k12.edu

Instructor and Staff Format: @townshipschools.k12.edu
jane.apples@townshipschools.k12.edu

- Student Progress: Disabled
- Gradebook & Classes: Disabled
Apple School Manager
Admin portal

Create managed Apple IDs
Apple School Manager
Admin portal

Create managed Apple IDs
Create rosters
Apple School Manager
Admin portal

Create managed Apple IDs
Create rosters
Manage content and devices
Apple School Manager
Admin portal

Create managed Apple IDs
Create rosters
Manage content and devices
Opt in for progress reporting
Apple School Manager
Admin portal

Create managed Apple IDs
Create rosters
Manage content and devices
Opt in for progress reporting
Purchase apps in bulk
Apple School Manager
Admin portal

Create managed Apple IDs
Create rosters
Manage content and devices
Opt in for progress reporting
Purchase apps in bulk
Schoolwork
A new education app for students and teachers
Schoolwork
A new education app for students and teachers

View assigned handouts
Schoolwork
A new education app

View assigned handouts
Create new handouts
• Handout is a collection of tasks
Schoolwork
A new education app

View assigned handouts

Create new handouts
• Handout is a collection of tasks

View progress reports
Lifecycle of a Handout
Lifecycle of a Handout

Define assignable content
Lifecycle of a Handout
Define assignable content

Content is represented via CLSContext
• Forms a tree structure
Lifecycle of a Handout
Define assignable content

Content is represented via `CLSContext`
- Forms a tree structure

One main app context (root context)
- Created by default
Lifecycle of a Handout
Define assignable content

Content is represented via CLSContext
• Forms a tree structure

One main app context (root context)
• Created by default

App content is added to the app context
Lifecycle of a Handout
Define assignable content

Content is represented via **CLSContext**
• Forms a tree structure
One main **app context** (root context)
• Created by default
App **content** is added to the **app context**
Define content as early as possible
Lifecycle of a Handout

Assignment

myReaderApp - Teacher iPad

myReaderApp - Student iPad
Lifecycle of a Handout

Assignment

myReaderApp - Teacher iPad

myReaderApp - Student iPad
Lifecycle of a Handout

Assignment

myReaderApp - Teacher iPad

myReaderApp - Student iPad
Lifecycle of a Handout

Assignment

myReaderApp - Teacher iPad

myReaderApp - Student iPad
Lifecycle of a Handout

Assignment

myReaderApp - Teacher iPad

myReaderApp - Student iPad
Lifecycle of a Handout

Assignment

myReaderApp - Teacher iPad

myReaderApp - Student iPad
Lifecycle of a Handout

Assignment

myReaderApp - Teacher iPad

myReaderApp - Student iPad
Developer Mode
Settings > Developer > ClassKit API
Developer Mode
Settings > Developer > ClassKit API

ACT AS OTHER ROLE
- Off
- Teacher
- Student

Reset Development Data
Framework Classes
Framework Classes

CLSDataStore

Data store
CLSDataStore
Maintains the context tree
CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property
CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property

Keeps track of modified objects
CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property

Keeps track of modified objects

• Objects saved by calling `CLSDataStore.save(completion:)`
CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property

Keeps track of modified objects

- Objects saved by calling `CLSDataStore.save(completion:)`

One shared instance
Framework Classes

CLSDataStore

Data store

Data Store
Framework Classes

CLSContext

Data store

Main app context

App defined contexts
Context Identity
Context Identity

identifier property
Context Identity

- **identifier** property
- Unique across siblings
Context Identity

identifier property
• Unique across siblings

Context Identifier Path
Context Identity

**identifier** property

- Unique across siblings

Context Identifier Path

- Identifies a context within a context tree
Context Identity

**Identifier** property
- Unique across siblings

Context Identifier Path
- Identifies a context within a context tree
- Array of context identifiers
Context Identifier Path

["com.company.app", "book", "chapter-2", "section-1"]
Context Identifier Path

["com.company.app", "book", "chapter-2", "section-1"]
Context Identifier Path

["com.company.app", "book", "chapter-2", "section-1"]
Context Identifier Path

["com.company.app", "book", "chapter-2", "section-1"]
Context Identifier Path

["com.company.app", "book", "chapter-2", "section-1"]
let path = ["com.company.app", "book-1", "chapter-2", "section-1"]
CLSDataStore.shared.contexts(matchingIdentifierPath: path) { contextsAmongPath, error in
    // process contexts among path
}
Context Lookup
Using relative context path

```swift
let path = ["chapter-2", "section-1"]
bookContext.descendant(matchingIdentifierPath: path) { context, error in
    // use section1 context here
}
```
Context Lookup
Using predicates

```swift
let predicate = NSPredicate(format: "parent = %@", bookContext)
CLSDataStore.shared.contexts(matching: predicate) { contexts, error in
    // process all chapter contexts
}
```
protocol CLSDataStoreDelegate {
    func createContext(forIdentifier identifier: String,
                        parentContext: CLSContext,
                        parentIdentifierPath: [String]) -> CLSContext?
}
Works in conjunction with path query methods

```swift
protocol CLSDataStoreDelegate {
    func createContext(forIdentifier identifier: String,
                        parentContext: CLSContext,
                        parentIdentifierPath: [String]) -> CLSContext?
}
```
CLSDataStoreDelegate

Works in conjunction with path query methods

Used to create missing contexts on demand

```swift
protocol CLSDataStoreDelegate {
    func createContext(forIdentifier identifier: String,
                        parentContext: CLSContext,
                        parentIdentifierPath: [String]) -> CLSContext?
}
```
Works in conjunction with path query methods

Used to create missing contexts on demand

Useful for apps with dynamic content

```swift
protocol CLSDataStoreDelegate {
    func createContext(forIdentifier identifier: String,
                        parentContext: CLSContext,
                        parentIdentifierPath: [String]) -> CLSContext?;
}
```
Delegate in Action
Defining section 1

"com.company.app", "book", "chapter-2", "section-1"
Delegate in Action
Defining section 1

["com.company.app", "book", "chapter-2", "section-1"]
Delegate in Action
Defining section 1

["com.company.app", "book", "chapter-2", "section-1"]
Delegate in Action

Defining section 1

["com.company.app", "book", "chapter-2", "section-1"]
Delegate in Action

Defining section 1

["com.company.app", "book", "chapter-2", "section-1"]
Delegate in Action

Defining section 1

["com.company.app", "book", "chapter-2", "section-1"]
Delegate in Action

Defining section 1

["com.company.app", "book", "chapter-2", "section-1"]
Delegate in Action
Defining section 1

["com.company.app", "book", "chapter-2", "section-1"]
Object Graph

CLSContext

Data store

Main app context

App defined contexts
Object Graph
CLSActivity

Data store
Main app context
App defined contexts
Activity

Diagram:
- Data Store
- App
  - Context
  - Context
- Activity
CLSActivity

Always associated with contexts
CLSActivity

Always associated with contexts

Created by calling `CLSContext.createNewActivity()` → `CLSActivity`
CLSActivity

Always associated with contexts

Created by calling `CLSContext.createNewActivity() -> CLSActivity`

Access current using `CLSContext.currentActivity: CLSActivity?`
Always associated with contexts

Created by calling `CLSContext.createNewActivity() -> CLSActivity`

Access current using `CLSContext.currentActivity: CLSActivity?`

Creating a new activity is akin to a new attempt
Activity

Adding progress

// adding progress
activity.progress = 0.5 // 50%

// or
activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
// adding progress

activity.progress = 0.5 // 50%

// or

activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
// adding progress
activity.progress = 0.5 // 50%

// or
activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
// adding progress
activity.progress = 0.5 // 50%

// or
activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
Object Graph

Data store
Main app context
App defined contexts
Activity
Object Graph

Data store
Main app context
App defined contexts
Activity
Primary activity item
Additional activity items
CLSActivityItem
Abstract class with three subclasses
**CLSActivityItem**
Abstract class with three subclasses

**CLSQuantityItem**
• Used for scalar values, (e.g. 5 Hints)
**CLSActivityItem**
Abstract class with three subclasses

**CLSQuantityItem**
- Used for scalar values, (e.g. 5 Hints)

**CLSScoreItem**
- Used for X out of Y values (e.g. 5 out 7 questions answered)
CLSActivityItem
Abstract class with three subclasses

CLSQuantityItem
• Used for scalar values, (e.g. 5 Hints)

CLSScoreItem
• Used for X out of Y values (e.g. 5 out 7 questions answered)

CLSBinaryItem
• Used for binary values
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
Adding activity items

// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
Adding activity items

// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
Adding activity items

// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
Adding activity items

// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
Best practices
Best practices

Always set the same subclass for primary activity items
CLSActivityItem

Best practices

Always set the same subclass for primary activity items

Provide clear and concise titles
Best practices

Always set the same subclass for primary activity items

Provide clear and concise titles

Make use of additional activity items
Adopting ClassKit

Marin Eubanks, ClassKit team
Select which difficulty level you would like to play.

Addition Quiz

Multiplication Quiz
View score board or start quiz.

View Score Board

Start Quiz
x + 6 = 10
100%
Steps for Adopting ClassKit
Steps for Adopting ClassKit

Define **CLSContext** structure
Steps for Adopting ClassKit

Define **CLSContext** structure

Add **CLSActivity** and **CLSActivityItems**

![Diagram]

- Data Store
- App
- Quiz
  - Addition
  - Multiplication
  - Activity
  - Primary Item
  - Additional Item
Steps for Adopting ClassKit

Define **CLSContext** structure

Add **CLSActivity** and **CLSActivityItems**

Support deep linking

```swift
func application(_ application: UIApplication,
continue userActivity: NSUserActivity,
restorationHandler: @escaping ([UIUserActivityRestoring]?) -> Void) -> Bool
```
Steps for Adopting ClassKit

Define **CLSContext** structure

Add **CLSActivity** and **CLSActivityItems**

Support deep linking

Test with Developer Mode and Schoolwork
Defining CLSContext Tree

Quiz
Addition
Quiz
Score Board
Multiplication
Quiz
Score Board

Quizzler.app
Defining CLSContext Tree

[Diagram showing a tree structure with nodes labeled as 'App', 'Addition', 'Quiz', 'Multiplication', and 'Quiz']

Quizzler.app
Defining CLSContext Tree

App

Addition Math Quiz

Multiplication Math Quiz

Quizzler.app
Use clear titles
CLSContext

Use clear titles
Define contexts early
CLSCContext

Use clear titles
Define contexts early
Take advantage of displayOrder
Addition Quiz Handout
Complete the math quiz on addition.

0 NOT DONE  1 DONE

Addition Math Quiz
Quiz Quizzer

AVG. SCORE  AVG. TOTAL TIME  DONE
100%  2 min  1/1

STUDENTS  SCORE  TOTAL TIME  DONE
JA  Johnny Appleseed  100%  2 min  ✓
**Addition Quiz Handout**
Complete the math quiz on addition.

<table>
<thead>
<tr>
<th>DEV CLASS</th>
<th>Addition Math Quiz</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Quiz</td>
</tr>
<tr>
<td></td>
<td>izzler</td>
</tr>
</tbody>
</table>

**Students**

<table>
<thead>
<tr>
<th></th>
<th>Score</th>
<th>Total Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Johnny Appleseed</td>
<td>100%</td>
<td>2 min</td>
</tr>
</tbody>
</table>
// create activity
let activity = quizContext.createNewActivity()
activity.start()
//...
activity.stop()
// create activity
let activity = quizContext.createNewActivity()
activity.start()
// ...
activity.stop()
Addition Quiz Handout
Complete the math quiz on addition.

Addition Math Quiz
Quiz: Quizizzler

AVG. SCORE: 100%
AVG. TOTAL TIME: 2 min
DONE: 1/1

Students
Johnny Appleseed
Score: 100%
Total Time: 2 min
DONE
Addition Quiz Handout
Complete the math quiz on addition.

0 NOT DONE 1 DONE

Addition Math Quiz
Quiz Quizzler

AVG. SCORE 100%
AVG. TOTAL TIME 2 min
DONE 1 / 1

STUDENTS  SCORE  TOTAL TIME  DONE
JA  Johnny Appleseed  100%  2 min  ✔
// create activity
let activity = quizContext.createNewActivity()

// adding primary activity items
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score
// create activity
let activity = quizContext.createNewActivity()
// adding primary activity items
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score
Addition Quiz Handout
Complete the math quiz on addition.

Score: 100%
Total Time: 2 min
Done: Yes
Passed: Passed
Passed: Passed
Passed: Passed
Addition Quiz Handout
Complete the math quiz on addition.

0 NOT DONE 1 DONE

Addition Math Quiz
Quiz Quizzer

Johnny Appleseed

Score 100%
Class Avg.: 100%

Total Time 2 min
Class Avg.: 2 min

Done Yes

Passed | Not Passed
----------
Passed
Passed
Passed
// create activity
let activity = quizContext.createNewActivity()

let binaryItem = CLSBinaryItem(identifier: "question_1", title: "Question 1", type:.passFail)
binaryItem.value = true

activity.addAdditionalActivityItem(binaryItem)
// create activity
let activity = quizContext.createNewActivity()

let binaryItem = CLSBinaryItem(identifier: "question_1", title: "Question 1", type:.passFail)
binaryItem.value = true

activity.addAdditionalActivityItem(binaryItem)
Summary

General Overview

Framework Classes

Adopting ClassKit

Testing via Developer Mode and Schoolwork
Summary
ClassKit best practices

Declare contexts early

Not everything needs to be a context

Take advantage of CLSDataStoreDelegate

Provide extra information via activity items
Summary

General best practices for education apps

Remove StoreKit dependence

Support purgeable storage

Implement settings access via Managed App Config

More information at: https://developer.apple.com/education/
More Information


<table>
<thead>
<tr>
<th>ClassKit and Education Technologies Lab</th>
<th>Technology Lab 4</th>
<th>Wednesday 10:45AM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClassKit and Education Technologies Lab</td>
<td>Technology Lab 2</td>
<td>Thursday 09:00AM</td>
</tr>
</tbody>
</table>