

#WWDC19

In-App Purchases and Using Server-to-Server Notifications

Dana DuBois, App Store Engineering Manager
Tori Shurman, App Store Commerce Engineer
Manjeet Chawla, App Store Commerce

What's new in StoreKit

Server-to-server notifications

Subscription lifecycle

Reducing involuntary churn

What's new in StoreKit

Server-to-server notifications

Subscription lifecycle

Reducing involuntary churn

What's new in StoreKit

Server-to-server notifications

Subscription lifecycle

Reducing involuntary churn

What's new in StoreKit

Server-to-server notifications

Subscription lifecycle

Reducing involuntary churn

What's new in StoreKit

Server-to-server notifications

Subscription lifecycle

Reducing involuntary churn

Subscription Offers

Subscription Offers

Introduced in iOS 12.2, tvOS 12.2, macOS 10.14.4

Subscription Offers

Introduced in iOS 12.2, tvOS 12.2, macOS 10.14.4

Retaining and winning back subscribers

Subscription Offers

Introduced in iOS 12.2, tvOS 12.2, macOS 10.14.4

Retaining and winning back subscribers

10 active offers per subscription

Subscription Offers

Introduced in iOS 12.2, tvOS 12.2, macOS 10.14.4

Retaining and winning back subscribers

10 active offers per subscription

Your app decides what offers to present

Subscription Offers

Introduced in iOS 12.2, tvOS 12.2, macOS 10.14.4

Retaining and winning back subscribers

10 active offers per subscription

Your app decides what offers to present

Subscription Offers Best Practices

Friday, 2:00

In-App Purchases and Subscriptions Lab

Friday, 3:00

SKStorefront

SKStorefront



NEW

SKStorefront



NEW

Present the right content to your customers

SKStorefront



NEW

Present the right content to your customers

Exposes the region that the App Store is currently set to

SKStorefront



NEW

Present the right content to your customers

Exposes the region that the App Store is currently set to

Device specific and can change over time

```
// Getting the storefront

import StoreKit
import os.log

if let sf = SKPaymentQueue.default().storefront {
    os_log("Storefront ISO 3 Country Code: %@", sf.countryCode)
}
```

```
// Getting the storefront

import StoreKit
import os.log

if let sf = SKPaymentQueue.default().storefront {
    os_log("Storefront ISO 3 Country Code: %@", sf.countryCode)
}
```

```
// Getting the storefront

import StoreKit
import os.log

if let sf = SKPaymentQueue.default().storefront {
    os_log("Storefront ISO 3 Country Code: %@", sf.countryCode)
}
```

```
// Getting the storefront

import StoreKit
import os.log

if let sf = SKPaymentQueue.default().storefront {
    os_log("Storefront ISO 3 Country Code: %@", sf.countryCode)
}
```

```
// Using the storefront

func shouldShow(_ productId: String, in storefront: SKStorefront) -> Bool {
    var rv = true

    //myProducts is a dictionary representing your own metadata for products
    //keyed on an SKProduct.productId
    if let myProduct = myProducts[productId] {
        rv = myProduct.countryCodes.contains(storefront.countryCode)
    }

    return rv
}
```

```
// Using the storefront

func shouldShow(_ productId: String, in storefront: SKStorefront) -> Bool {
    var rv = true

    //myProducts is a dictionary representing your own metadata for products
    //keyed on an SKProduct.productId
    if let myProduct = myProducts[productId] {
        rv = myProduct.countryCodes.contains(storefront.countryCode)
    }

    return rv
}
```

```
// Using the storefront (cont.)

func fetchProductInfo() {
    var identifiers = Set<String>()
    if let storefront = SKPaymentQueue.default().storefront {
        for (identifier, _) in myProducts {
            if shouldShow(identifier, in: storefront) {
                identifiers.insert(identifier)
            }
        }
        let request = SKProductsRequest(productIdentifiers: identifiers)
        request.delegate = self
        request.start()
    }
}
```



```
// Listening for storefront changes

//MARK: - SKPaymentTransactionObserver methods
func paymentQueueDidChangeStorefront(_ queue: SKPaymentQueue) {
    if let storefront = queue.storefront {
        //refresh the displayed products based on the new storefront
        for product in storeProducts {
            if shouldShow(product.productId, in: storefront) {
                //display this product in your store UI
            }
        }
    }
}
}
```

```
// Listening for storefront changes

//MARK: - SKPaymentTransactionObserver methods
func paymentQueueDidChangeStorefront(_ queue: SKPaymentQueue) {
    if let storefront = queue.storefront {
        //refresh the displayed products based on the new storefront
        for product in storeProducts {
            if shouldShow(product.productId, in: storefront) {
                //display this product in your store UI
            }
        }
    }
}
}
```

```
// Listening for storefront changes

//MARK: - SKPaymentTransactionObserver methods
func paymentQueueDidChangeStorefront(_ queue: SKPaymentQueue) {
    if let storefront = queue.storefront {
        //refresh the displayed products based on the new storefront
        for product in storeProducts {
            if shouldShow(product.productId, in: storefront) {
                //display this product in your store UI
            }
        }
    }
}
}
```

```
// Listening for storefront changes

//MARK: - SKPaymentTransactionObserver methods
func paymentQueueDidChangeStorefront(_ queue: SKPaymentQueue) {
    if let storefront = queue.storefront {
        //refresh the displayed products based on the new storefront
        for product in storeProducts {
            if shouldShow(product.productId, in: storefront) {
                //display this product in your store UI
            }
        }
    }
}
}
```

```
// Responding to storefront changes during purchase

SKPaymentQueue.default().delegate = self //set your object as the SKPaymentQueue delegate

//MARK: - SKPaymentQueueDelegate methods
func paymentQueue(_ paymentQueue: SKPaymentQueue,
                  shouldContinue transaction: SKPaymentTransaction,
                  in newStorefront: SKStorefront) -> Bool {
    return shouldShow(transaction.payment.productId, in: newStorefront)
}
```

```
// Responding to storefront changes during purchase
```

```
SKPaymentQueue.default().delegate = self //set your object as the SKPaymentQueue delegate
```

```
//MARK: – SKPaymentQueueDelegate methods
```

```
func paymentQueue(_ paymentQueue: SKPaymentQueue,  
                 shouldContinue transaction: SKPaymentTransaction,  
                 in newStorefront: SKStorefront) -> Bool {  
    return shouldShow(transaction.payment.productId, in: newStorefront)  
}
```

```
// Responding to storefront changes during purchase

SKPaymentQueue.default().delegate = self //set your object as the SKPaymentQueue delegate

//MARK: - SKPaymentQueueDelegate methods
func paymentQueue(_ paymentQueue: SKPaymentQueue,
                  shouldContinue transaction: SKPaymentTransaction,
                  in newStorefront: SKStorefront) -> Bool {
    return shouldShow(transaction.payment.productId, in: newStorefront)
}
```

```
// Responding to storefront changes during purchase

//MARK: - SKPaymentTransactionObserver methods
func paymentQueue(_ queue: SKPaymentQueue,
                  updatedTransactions transactions: [SKPaymentTransaction]) {
    for transaction in transactions {
        if let transactionError = transaction.error as NSError? {
            if transactionError.domain == SKErrorDomain
                && transactionError.code == SKError.storeProductNotAvailable.rawValue {
                //Show an alert
            }
        }
    }
}
}
```



```
// Responding to storefront changes during purchase

//MARK: - SKPaymentTransactionObserver methods
func paymentQueue(_ queue: SKPaymentQueue,
                  updatedTransactions transactions: [SKPaymentTransaction]) {
    for transaction in transactions {
        if let transactionError = transaction.error as NSError? {
            if transactionError.domain == SKErrorDomain
                && transactionError.code == SKError.storeProductNotAvailable.rawValue {
                //Show an alert
            }
        }
    }
}
}
```

```
// Responding to storefront changes during purchase

//MARK: - SKPaymentTransactionObserver methods
func paymentQueue(_ queue: SKPaymentQueue,
                  updatedTransactions transactions: [SKPaymentTransaction]) {
    for transaction in transactions {
        if let transactionError = transaction.error as NSError? {
            if transactionError.domain == SKErrorDomain
                && transactionError.code == SKError.storeProductNotAvailable.rawValue {
                //Show an alert
            }
        }
    }
}
}
```

App Pre-Orders

App Pre-Orders

App Pre-Orders

Introduced in iOS 11.2, tvOS 11.2, and macOS 10.13.2

App Pre-Orders

Introduced in iOS 11.2, tvOS 11.2, and macOS 10.13.2

Now available in the Watch App Store with watchOS 6.0

App Pre-Orders

Introduced in iOS 11.2, tvOS 11.2, and macOS 10.13.2

Now available in the Watch App Store with watchOS 6.0

Coming Soon available in the receipt

Server-to-Server Notifications

Tori Shurman, App Store Commerce Engineer

What Are Server-to-Server Notifications?

What Are Server-to-Server Notifications?

HTTP POST with a JSON body

What Are Server-to-Server Notifications?

HTTP POST with a JSON body

Previously known as statusUpdateNotifications

What Are Server-to-Server Notifications?

HTTP POST with a JSON body

Previously known as statusUpdateNotifications

Return a 200 response from your endpoint to indicate a received message

What Are Server-to-Server Notifications?

HTTP POST with a JSON body

Previously known as statusUpdateNotifications

Return a 200 response from your endpoint to indicate a received message

Will retry up to three times

Setting Up Your Endpoint

The screenshot shows the 'App Information' page in the Apple App Store Connect interface. The page is for the app 'Forest Explorer' by 'Apple Inc.'. The left sidebar shows 'App Information' as the selected section under 'APP STORE INFORMATION'. The main content area is divided into three sections: 'App Information', 'Localizable Information', and 'General Information'. The 'App Information' section contains a 'Save' button and a note that changes will be released with the next app version. The 'Localizable Information' section includes a 'Subtitle' field with the placeholder 'Optional' and an 'Apple TV Privacy Policy' field with the text 'Apple is committed to your privacy. Read our customer Privacy Policy for a clear explanation of how we collect, use, disclose, transfer, and store your information.' The 'General Information' section includes fields for 'Bundle ID' (com.mv2.ustwolnc), 'SKU' (Forest Explorer), 'Apple ID' (12734021831), 'Primary Language' (English), 'Category' (Entertainment), 'License Agreement' (Custom), 'Rating' (Ages 4+), and 'Subscription Status URL' (http://www.2bnffa3k.execute-api.us-east-1.apple.com/ds/).

App Store Connect My Apps

Apple Inc. Forest Explorer

John Appleseed Nature Lab

App Store Features TestFlight Activity App Analytics Sales and Trends

APP STORE INFORMATION

App Information Pricing and Availability

iOS APP

1.0 Prepare for Submiss...

+ VERSION OR PLATFORM

App Information

This information is used for all platforms of this app. Any changes will be released with your next app version.

Save

Localizable Information

Subtitle ?

Optional

Apple TV Privacy Policy ?

Apple is committed to your privacy. Read our customer Privacy Policy for a clear explanation of how we collect, use, disclose, transfer, and store your information.

General Information

Bundle ID ?

com.mv2.ustwolnc

SKU ?

Forest Explorer

Apple ID ?

12734021831

Primary Language ?

English

Category ?

Entertainment

License Agreement [Edit](#)

Custom

Rating ?

Ages 4+ [Additional Ratings](#)

Subscription Status URL ?

http://www.2bnffa3k.execute-api.us-east-1.apple.com/ds/

Setting Up Your Endpoint

The screenshot shows the 'App Information' page in the Apple App Store Connect interface. The page is for the app 'Forest Explorer' by 'Nature Lab'. The left sidebar shows 'App Information' selected under 'APP STORE INFORMATION'. The main content area is divided into sections: 'App Information', 'Localizable Information', 'Apple TV Privacy Policy', and 'General Information'. The 'Subscription Status URL' field is highlighted with a red box.

App Store Connect My Apps ▾ Apple Inc. Forest Explorer ▾ John Appleseed ▾ Nature Lab ?

App Store Features TestFlight Activity App Analytics Sales and Trends

APP STORE INFORMATION

- App Information
- Pricing and Availability

iOS APP

- 1.0 Prepare for Submiss...

+ VERSION OR PLATFORM

App Information

This information is used for all platforms of this app. Any changes will be released with your next app version. Save

Localizable Information

Subtitle ?
Optional

Apple TV Privacy Policy ?
Apple is committed to your privacy. Read our customer Privacy Policy for a clear explanation of how we collect, use, disclose, transfer, and store your information.

General Information

Bundle ID ?
com.mv2.ustwolnc

SKU ?
Forest Explorer

Apple ID ?
12734021831

Primary Language ?
English

Category ?
Entertainment

License Agreement [Edit](#)
Custom

Rating ?
Ages 4+ [Additional Ratings](#)

Subscription Status URL ?
`http://www.2bnffa3k.execute-api.us-east-1.apple.com/ds/`

Setting Up Your Server

Setting Up Your Server

The connection must be App Transport Security (ATS) compliant

Setting Up Your Server

The connection must be App Transport Security (ATS) compliant
Certificate issued by a trusted certificate authority

Setting Up Your Server

The connection must be App Transport Security (ATS) compliant

Certificate issued by a trusted certificate authority

TLS 1.2

Setting Up Your Server

The connection must be App Transport Security (ATS) compliant

Certificate issued by a trusted certificate authority

TLS 1.2

Use the AES-128 or AES-256 symmetric cipher

Setting Up Your Server

The connection must be App Transport Security (ATS) compliant

Certificate issued by a trusted certificate authority

TLS 1.2

Use the AES-128 or AES-256 symmetric cipher

Signed and hashed using SHA-256 or greater

What's New with Server-to-Server Notifications

Unified Receipt

Unified Receipt

NEW

Unified Receipt



NEW

Contains the history of in-app purchases

Unified Receipt

NEW

Contains the history of in-app purchases

Previously, only available through `/verifyReceipt`

Unified Receipt



NEW

Contains the history of in-app purchases

Previously, only available through `/verifyReceipt`

New field in notification

Unified Receipt



NEW

Contains the history of in-app purchases

Previously, only available through `/verifyReceipt`

New field in notification

- `unified_receipt`

Unified Receipt

NEW

Contains the history of in-app purchases

Previously, only available through `/verifyReceipt`

New field in notification

- `unified_receipt`

`latest_receipt` and `latest_receipt_info` no longer needed

Not tied to a specific install of an app

Not tied to a specific install of an app
Should always be stored on your server

Unified Receipt

Contents

NEW

Unified Receipt

Contents



`unified_receipt_info` contents

Unified Receipt

Contents

NEW

`unified_receipt_info` contents

- `latest_receipt`

Unified Receipt

Contents

NEW

`unified_receipt_info` contents

- `latest_receipt`
- `latest_receipt_info`

Unified Receipt

Contents

NEW

`unified_receipt_info` contents

- `latest_receipt`
- `latest_receipt_info`
- `pending_renewal_info`

Unified Receipt

Contents

NEW

`unified_receipt_info` contents

- `latest_receipt`
- `latest_receipt_info`
- `pending_renewal_info`
- `status`

Unified Receipt

Contents

NEW

`unified_receipt_info` contents

- `latest_receipt`
- `latest_receipt_info`
- `pending_renewal_info`
- `status`
- `environment`

Unified Receipt

Contents

NEW

`unified_receipt_info` contents

- `latest_receipt`
- `latest_receipt_info`
- `pending_renewal_info`
- `status`
- `environment`

Limited to the 100 latest In App Purchases

Notification Types

Notification Types

INITIAL_BUY

INTERACTIVE_RENEWAL

DID_CHANGE_RENEWAL_PREF

CANCEL

DID_CHANGE_RENEWAL_STATUS

DID_FAIL_TO_RENEW

DID_RECOVER

PRICE_INCREASE_CONSENT

Notification Types

INITIAL_BUY

INTERACTIVE_RENEWAL

DID_CHANGE_RENEWAL_PREF

CANCEL

DID_CHANGE_RENEWAL_STATUS NEW

DID_FAIL_TO_RENEW

DID_RECOVER

PRICE_INCREASE_CONSENT

Notification Types

INITIAL_BUY

INTERACTIVE_RENEWAL

DID_CHANGE_RENEWAL_PREF

CANCEL

DID_CHANGE_RENEWAL_STATUS

DID_FAIL_TO_RENEW NEW

DID_RECOVER

PRICE_INCREASE_CONSENT

Notification Types

INITIAL_BUY

INTERACTIVE_RENEWAL

DID_CHANGE_RENEWAL_PREF

CANCEL

DID_CHANGE_RENEWAL_STATUS

DID_FAIL_TO_RENEW

DID_RECOVER NEW

PRICE_INCREASE_CONSENT

Notification Types

INITIAL_BUY

INTERACTIVE_RENEWAL

DID_CHANGE_RENEWAL_PREF

CANCEL

DID_CHANGE_RENEWAL_STATUS

DID_FAIL_TO_RENEW

DID_RECOVER

PRICE_INCREASE_CONSENT NEW

New Notification Types

New Notification Types

NEW

New Notification Types

NEW

Notification Type

Meaning

New Fields in Payload

Available Date

New Notification Types

NEW

Notification Type	Meaning	New Fields in Payload	Available Date
DID_CHANGE_RENEWAL_STATUS	Auto-renew toggled on/off		Now

New Notification Types

NEW

Notification Type	Meaning	New Fields in Payload	Available Date
DID_CHANGE_RENEWAL_STATUS	Auto-renew toggled on/off		Now
DID_FAIL_TO_RENEW	Auto-renew fails at first attempt to renew		Fall 2019

New Notification Types

NEW

Notification Type	Meaning	New Fields in Payload	Available Date
DID_CHANGE_RENEWAL_STATUS	Auto-renew toggled on/off		Now
DID_FAIL_TO_RENEW	Auto-renew fails at first attempt to renew		Fall 2019
DID_RECOVER	Subscription recovered in billing retry		Fall 2019

New Notification Types

NEW

Notification Type	Meaning	New Fields in Payload	Available Date
DID_CHANGE_RENEWAL_STATUS	Auto-renew toggled on/off		Now
DID_FAIL_TO_RENEW	Auto-renew fails at first attempt to renew		Fall 2019
DID_RECOVER	Subscription recovered in billing retry		Fall 2019
PRICE_INCREASE_CONSENT	User has entered a price increase flow	price_increase_effective_date	Fall 2019

Handling Notifications

Existing Notification Type Overview

Existing Notification Type Overview

Notification Type

Sent When

Fields to Look for in
JSON Payload

Existing Notification Type Overview

Notification Type	Sent When	Fields to Look for in JSON Payload
INITIAL_BUY	A subscription is first purchased	<code>purchase_date</code> <code>original_transaction_id</code> <code>web_order_line_item_id</code>

Existing Notification Type Overview

Notification Type	Sent When	Fields to Look for in JSON Payload
<code>INITIAL_BUY</code>	A subscription is first purchased	<code>purchase_date</code> <code>original_transaction_id</code> <code>web_order_line_item_id</code>
<code>INTERACTIVE_RENEWAL</code>	A subscription is renewed in the foreground	<code>purchase_date</code> <code>original_transaction_id</code> <code>web_order_line_item_id</code> <code>product_id</code>

Existing Notification Type Overview

Notification Type	Sent When	Fields to Look for in JSON Payload
<code>INITIAL_BUY</code>	A subscription is first purchased	<code>purchase_date</code> <code>original_transaction_id</code> <code>web_order_line_item_id</code>
<code>INTERACTIVE_RENEWAL</code>	A subscription is renewed in the foreground	<code>purchase_date</code> <code>original_transaction_id</code> <code>web_order_line_item_id</code> <code>product_id</code>
<code>DID_CHANGE_RENEWAL_PREF</code>	A customer downgrades	<code>auto_renew_product_id</code> <code>original_transaction_id</code>

Existing Notification Type Overview

Notification Type	Sent When	Fields to Look for in JSON Payload
INITIAL_BUY	A subscription is first purchased	<code>purchase_date</code> <code>original_transaction_id</code> <code>web_order_line_item_id</code>
INTERACTIVE_RENEWAL	A subscription is renewed in the foreground	<code>purchase_date</code> <code>original_transaction_id</code> <code>web_order_line_item_id</code> <code>product_id</code>
DID_CHANGE_RENEWAL_PREF	A customer downgrades	<code>auto_renew_product_id</code> <code>original_transaction_id</code>
CANCEL	Customer support issues a refund	<code>original_transaction_id</code> <code>cancellation_date</code> <code>product_id</code>

A Subscription is First Purchased

INITIAL_BUY

A Subscription is First Purchased

INITIAL_BUY

Provide service for a newly purchased subscription

A Subscription is First Purchased

INITIAL_BUY

Provide service for a newly purchased subscription

Update the customer's subscription status accordingly

A Subscription is First Purchased

INITIAL_BUY

Provide service for a newly purchased subscription

Update the customer's subscription status accordingly

Field Name

Type

Location

Example Value

Meaning

A Subscription is First Purchased

INITIAL_BUY

Provide service for a newly purchased subscription

Update the customer's subscription status accordingly

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase

A Subscription is First Purchased

INITIAL_BUY

Provide service for a newly purchased subscription

Update the customer's subscription status accordingly

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription

A Subscription is First Purchased

INITIAL_BUY

Provide service for a newly purchased subscription

Update the customer's subscription status accordingly

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription
<code>web_order_line_item_id</code>	String	latest_receipt_info	1234567	Links notification to verifyReceipt entry

A Subscription is First Purchased

INITIAL_BUY

Provide service for a newly purchased subscription

Update the customer's subscription status accordingly

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription
<code>web_order_line_item_id</code>	String	latest_receipt_info	1234567	Links notification to verifyReceipt entry
<code>product_id</code>	String	latest_receipt_info	my-subscription	Exact product customer has subscribed to

Renewal in the Foreground

INTERACTIVE_RENEWAL

Renewal in the Foreground

INTERACTIVE_RENEWAL

A customer upgrades — INTERACTIVE_RENEWAL + CANCEL

Renewal in the Foreground

INTERACTIVE_RENEWAL

A customer upgrades — INTERACTIVE_RENEWAL + CANCEL

A customer resubscribes after churn — only INTERACTIVE_RENEWAL

Renewal in the Foreground

INTERACTIVE_RENEWAL

A customer upgrades — INTERACTIVE_RENEWAL + CANCEL

A customer resubscribes after churn — only INTERACTIVE_RENEWAL

Field Name

Type

Location

Example Value

Meaning

Renewal in the Foreground

INTERACTIVE_RENEWAL

A customer upgrades — `INTERACTIVE_RENEWAL + CANCEL`

A customer resubscribes after churn — only `INTERACTIVE_RENEWAL`

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase

Renewal in the Foreground

INTERACTIVE_RENEWAL

A customer upgrades — `INTERACTIVE_RENEWAL + CANCEL`

A customer resubscribes after churn — only `INTERACTIVE_RENEWAL`

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription

Renewal in the Foreground

INTERACTIVE_RENEWAL

A customer upgrades — `INTERACTIVE_RENEWAL + CANCEL`

A customer resubscribes after churn — only `INTERACTIVE_RENEWAL`

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription
<code>web_order_line_item_id</code>	String	latest_receipt_info	1234567	Links notification to verifyReceipt entry

Renewal in the Foreground

INTERACTIVE_RENEWAL

A customer upgrades — `INTERACTIVE_RENEWAL + CANCEL`

A customer resubscribes after churn — only `INTERACTIVE_RENEWAL`

Field Name	Type	Location	Example Value	Meaning
<code>purchase_date_ms</code>	String	latest_receipt_info	1559930400	Date of initial subscription purchase
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription
<code>web_order_line_item_id</code>	String	latest_receipt_info	1234567	Links notification to verifyReceipt entry
<code>product_id</code>	String	latest_receipt_info	my-subscription	Product the customer upgraded/resubscribed to

Subscription Downgrade

DID_CHANGE_RENEWAL_PREF

Subscription Downgrade

DID_CHANGE_RENEWAL_PREF

Update customer's subscription to a lower tier

Subscription Downgrade

DID_CHANGE_RENEWAL_PREF

Update customer's subscription to a lower tier

Field Name

Type

Location

Example Value

Meaning

Subscription Downgrade

DID_CHANGE_RENEWAL_PREF

Update customer's subscription to a lower tier

Field Name	Type	Location	Example Value	Meaning
auto_renew_product_id	String	Top level JSON field	com.my.subscriptoin.iap	Product customer will auto renew at

Subscription Downgrade

DID_CHANGE_RENEWAL_PREF

Update customer's subscription to a lower tier

Field Name	Type	Location	Example Value	Meaning
<code>auto_renew_product_id</code>	String	Top level JSON field	<code>com.my.subscriptoin.iap</code>	Product customer will auto renew at
<code>original_transaction_id</code>	String	<code>latest_receipt_info</code>	<code>1234567</code>	Unique identifier for subscription

Customer Support Issues a Refund

CANCEL

Customer Support Issues a Refund

CANCEL

Customer cancels through customer support — CANCEL

Customer Support Issues a Refund

CANCEL

Customer cancels through customer support — CANCEL

Customer upgrades subscription — INTERACTIVE_RENEWAL + CANCEL

Customer Support Issues a Refund

CANCEL

Customer cancels through customer support — CANCEL

Customer upgrades subscription — INTERACTIVE_RENEWAL + CANCEL

Field Name

Type

Location

Example Value

Meaning

Customer Support Issues a Refund

CANCEL

Customer cancels through customer support — CANCEL

Customer upgrades subscription — INTERACTIVE_RENEWAL + CANCEL

Field Name	Type	Location	Example Value	Meaning
cancellation_date_ms	String	Top level JSON field	1559930400	Date/time of cancellation

Customer Support Issues a Refund

CANCEL

Customer cancels through customer support — CANCEL

Customer upgrades subscription — INTERACTIVE_RENEWAL + CANCEL

Field Name	Type	Location	Example Value	Meaning
cancellation_date_ms	String	Top level JSON field	1559930400	Date/time of cancellation
original_transaction_id	String	latest_receipt_info	1234567	Unique identifier for subscription

Customer Support Issues a Refund

CANCEL

Customer cancels through customer support — CANCEL

Customer upgrades subscription — INTERACTIVE_RENEWAL + CANCEL

Field Name	Type	Location	Example Value	Meaning
cancellation_date_ms	String	Top level JSON field	1559930400	Date/time of cancellation
original_transaction_id	String	latest_receipt_info	1234567	Unique identifier for subscription
product_id	String	latest_receipt_info	my-subscription	Product that was cancelled

New Notification Type Overview

New Notification Type Overview

NEW

Notification Type

Sent when

Fields to look for in
JSON payload

New Notification Type Overview

NEW

Notification Type	Sent when	Fields to look for in JSON payload
<code>DID_CHANGE_RENEWAL_STATUS</code>	The customer toggles auto-renew	<code>auto_renew_status_change</code> <code>auto_renew_status</code> <code>original_transaction_id</code> <code>product_id</code>

New Notification Type Overview

NEW

Notification Type	Sent when	Fields to look for in JSON payload
<code>DID_CHANGE_RENEWAL_STATUS</code>	The customer toggles auto-renew	<code>auto_renew_status_change</code> <code>auto_renew_status</code> <code>original_transaction_id</code> <code>product_id</code>
<code>DID_FAIL_TO_RENEW</code>	Auto-renew fails at the first attempt to renew	<code>original_transaction_id</code> <code>is_in_billing_retry_period</code> <code>expires_date</code>

New Notification Type Overview

NEW

Notification Type	Sent when	Fields to look for in JSON payload
<code>DID_CHANGE_RENEWAL_STATUS</code>	The customer toggles auto-renew	<code>auto_renew_status_change</code> <code>auto_renew_status</code> <code>original_transaction_id</code> <code>product_id</code>
<code>DID_FAIL_TO_RENEW</code>	Auto-renew fails at the first attempt to renew	<code>original_transaction_id</code> <code>is_in_billing_retry_period</code> <code>expires_date</code>
<code>DID_RECOVER</code>	A subscription is recovered in billing retry	<code>original_transaction_id</code> <code>purchase_date</code> <code>expires_date</code>

New Notification Type Overview

NEW

Notification Type	Sent when	Fields to look for in JSON payload
<code>DID_CHANGE_RENEWAL_STATUS</code>	The customer toggles auto-renew	<code>auto_renew_status_change</code> <code>auto_renew_status</code> <code>original_transaction_id</code> <code>product_id</code>
<code>DID_FAIL_TO_RENEW</code>	Auto-renew fails at the first attempt to renew	<code>original_transaction_id</code> <code>is_in_billing_retry_period</code> <code>expires_date</code>
<code>DID_RECOVER</code>	A subscription is recovered in billing retry	<code>original_transaction_id</code> <code>purchase_date</code> <code>expires_date</code>
<code>PRICE_INCREASE_CONSENT</code>	The customer enters a price increase flow	<code>original_transaction_id</code> <code>expires_date</code> <code>price_consent_status</code>

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS

NEW

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS



NEW

Update customer's subscription status to reflect changes

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS



NEW

Update customer's subscription status to reflect changes

Deploy retention strategies

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS

NEW

Update customer's subscription status to reflect changes

Deploy retention strategies

Field Name	Type	Location	Example Value	Meaning
------------	------	----------	---------------	---------

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS

NEW

Update customer's subscription status to reflect changes

Deploy retention strategies

Field Name	Type	Location	Example Value	Meaning
<code>auto_renew_status_change_date_ms</code>	String	Top level JSON field	1559930400	When status was changed

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS

NEW

Update customer's subscription status to reflect changes

Deploy retention strategies

Field Name	Type	Location	Example Value	Meaning
<code>auto_renew_status_change_date_ms</code>	String	Top level JSON field	1559930400	When status was changed
<code>auto_renew_status</code>	String	Top level JSON field	true/false	True if the customer will auto-renew, false otherwise

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS

NEW

Update customer's subscription status to reflect changes

Deploy retention strategies

Field Name	Type	Location	Example Value	Meaning
<code>auto_renew_status_change_date_ms</code>	String	Top level JSON field	1559930400	When status was changed
<code>auto_renew_status</code>	String	Top level JSON field	true/false	True if the customer will auto-renew, false otherwise
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription

Auto-Renew Status Change

DID_CHANGE_RENEWAL_STATUS

NEW

Update customer's subscription status to reflect changes

Deploy retention strategies

Field Name	Type	Location	Example Value	Meaning
<code>auto_renew_status_change_date_ms</code>	String	Top level JSON field	1559930400	When status was changed
<code>auto_renew_status</code>	String	Top level JSON field	true/false	True if the customer will auto-renew, false otherwise
<code>original_transaction_id</code>	String	latest_receipt_info	1234567	Unique identifier for subscription
<code>product_id</code>	String	latest_receipt_info	my-subscription	Product whose auto-renew status changed

Auto-Renew Fails at First Attempt to Renew

DID_FAIL_TO_RENEW

Auto-Renew Fails at First Attempt to Renew

DID_FAIL_TO_RENEW

Optionally choose to suspend service

Auto-Renew Fails at First Attempt to Renew

DID_FAIL_TO_RENEW

Optionally choose to suspend service

Update customer's subscription status to "inactive" or "billing-retry"

Auto-Renew Fails at First Attempt to Renew

DID_FAIL_TO_RENEW

NEW

Optionally choose to suspend service

Update customer's subscription status to "inactive" or "billing-retry"

Field Name

Type

Location

Example Value

Meaning

Auto-Renew Fails at First Attempt to Renew

DID_FAIL_TO_RENEW

NEW

Optionally choose to suspend service

Update customer's subscription status to "inactive" or "billing-retry"

Field Name	Type	Location	Example Value	Meaning
<code>is_in_billing_retry_period</code>	String	pending_renewal_info	0 or 1	Whether the App Store is trying to renew the subscription

Auto-Renew Fails at First Attempt to Renew

DID_FAIL_TO_RENEW

NEW

Optionally choose to suspend service

Update customer's subscription status to "inactive" or "billing-retry"

Field Name	Type	Location	Example Value	Meaning
<code>is_in_billing_retry_period</code>	String	<code>pending_renewal_info</code>	0 or 1	Whether the App Store is trying to renew the subscription
<code>original_transaction_id</code>	String	<code>latest_receipt_info</code>	1234567	Unique identifier for subscription

Auto-Renew Fails at First Attempt to Renew

DID_FAIL_TO_RENEW

NEW

Optionally choose to suspend service

Update customer's subscription status to "inactive" or "billing-retry"

Field Name	Type	Location	Example Value	Meaning
<code>is_in_billing_retry_period</code>	String	<code>pending_renewal_info</code>	0 or 1	Whether the App Store is trying to renew the subscription
<code>original_transaction_id</code>	String	<code>latest_receipt_info</code>	1234567	Unique identifier for subscription
<code>expires_date_ms</code>	String	<code>latest_receipt_info</code>	1559930400	Date the subscription expired

Subscription Recovered in Billing Retry

DID_RECOVER

Subscription Recovered in Billing Retry

DID_RECOVER

Replaces existing **RENEWAL** notification

Subscription Recovered in Billing Retry

DID_RECOVER

Replaces existing **RENEWAL** notification

Restore service for a recovered subscription

Subscription Recovered in Billing Retry

DID_RECOVER

Replaces existing **RENEWAL** notification

Restore service for a recovered subscription

Update customer's subscription status to "active"

Subscription Recovered in Billing Retry

DID_RECOVER

NEW

Replaces existing **RENEWAL** notification

Restore service for a recovered subscription

Update customer's subscription status to "active"

Field Name

Type

Location

Example Value

Meaning

Subscription Recovered in Billing Retry

DID_RECOVER

NEW

Replaces existing **RENEWAL** notification

Restore service for a recovered subscription

Update customer's subscription status to "active"

Field Name	Type	Location	Example Value	Meaning
purchase_date_ms	String	latest_receipt_info	1559930400	Date of recovery

Subscription Recovered in Billing Retry

DID_RECOVER

NEW

Replaces existing **RENEWAL** notification

Restore service for a recovered subscription

Update customer's subscription status to "active"

Field Name	Type	Location	Example Value	Meaning
purchase_date_ms	String	latest_receipt_info	1559930400	Date of recovery
original_transaction_id	String	latest_receipt_info	1234567	Unique identifier for subscription

Subscription Recovered in Billing Retry

DID_RECOVER

NEW

Replaces existing **RENEWAL** notification

Restore service for a recovered subscription

Update customer's subscription status to "active"

Field Name	Type	Location	Example Value	Meaning
purchase_date_ms	String	latest_receipt_info	1559930400	Date of recovery
original_transaction_id	String	latest_receipt_info	1234567	Unique identifier for subscription
expires_date_ms	String	latest_receipt_info	1559930400	Date new subscription will expire

Customer Enters a Price Increase Flow

PRICE_INCREASE_CONSENT

Customer Enters a Price Increase Flow

PRICE_INCREASE_CONSENT

Update user's status to "price increase"

Customer Enters a Price Increase Flow

PRICE_INCREASE_CONSENT

Update user's status to "price increase"

Deploy in-app messaging

Customer Enters a Price Increase Flow

PRICE_INCREASE_CONSENT

NEW

Update user's status to "price increase"

Deploy in-app messaging

Field Name	Type	Location	Example Value	Meaning
------------	------	----------	---------------	---------

Customer Enters a Price Increase Flow

PRICE_INCREASE_CONSENT

NEW

Update user's status to "price increase"

Deploy in-app messaging

Field Name	Type	Location	Example Value	Meaning
<code>price_consent_status</code>	String	pending_renewal_info	0 or 1, default 0	1 if the customer has already consented, 0 otherwise

Customer Enters a Price Increase Flow

PRICE_INCREASE_CONSENT

NEW

Update user's status to "price increase"

Deploy in-app messaging

Field Name	Type	Location	Example Value	Meaning
<code>price_consent_status</code>	String	<code>pending_renewal_info</code>	0 or 1, default 0	1 if the customer has already consented, 0 otherwise
<code>original_transaction_id</code>	String	<code>latest_receipt_info</code>	1234567	Unique identifier for subscription

Customer Enters a Price Increase Flow

PRICE_INCREASE_CONSENT

NEW

Update user's status to "price increase"

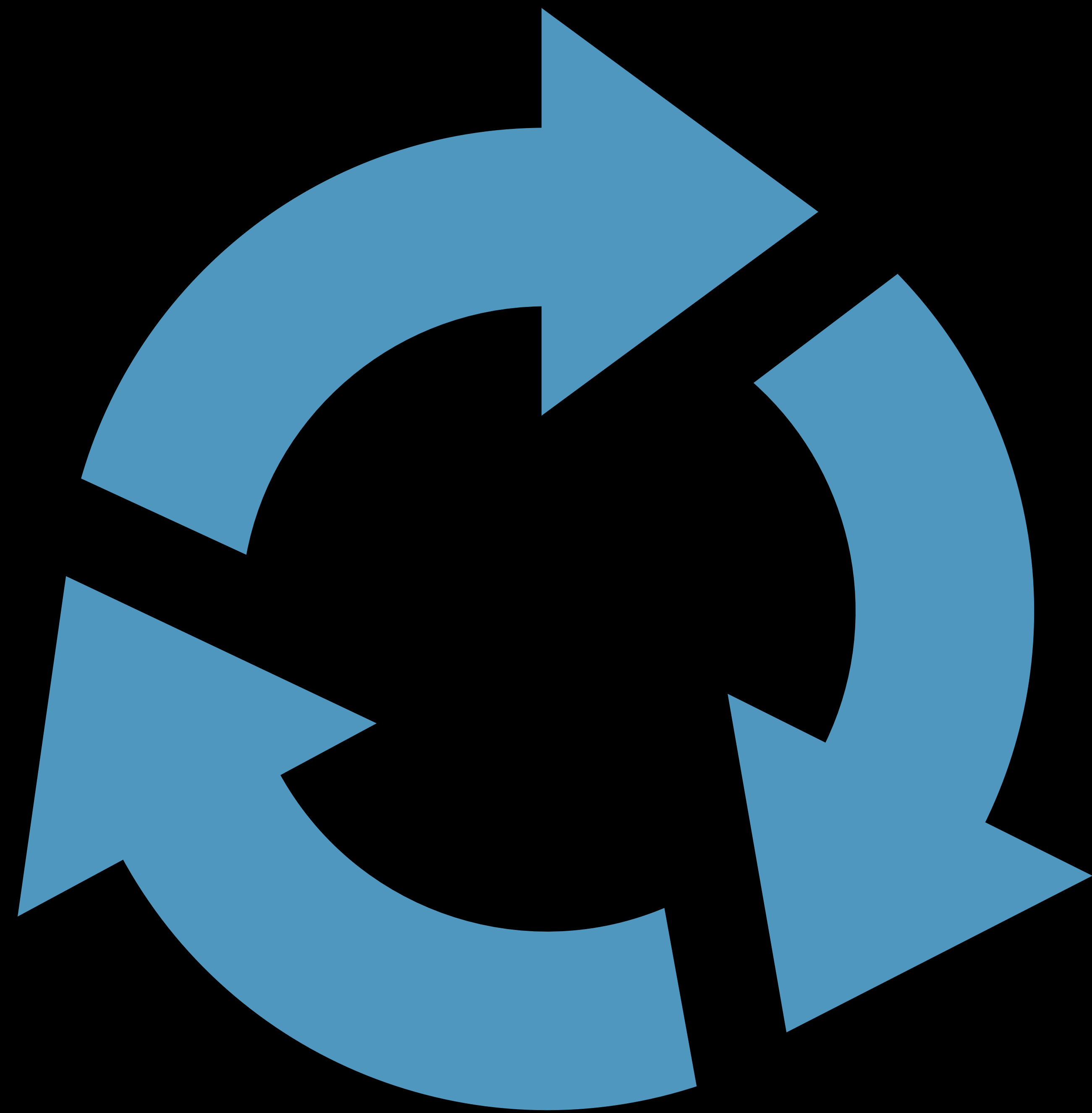
Deploy in-app messaging

Field Name	Type	Location	Example Value	Meaning
price_consent_status	String	pending_renewal_info	0 or 1, default 0	1 if the customer has already consented, 0 otherwise
original_transaction_id	String	latest_receipt_info	1234567	Unique identifier for subscription
expires_date_ms	String	latest_receipt_info	1559930400	Date new subscription will expire

Subscription Lifecycle

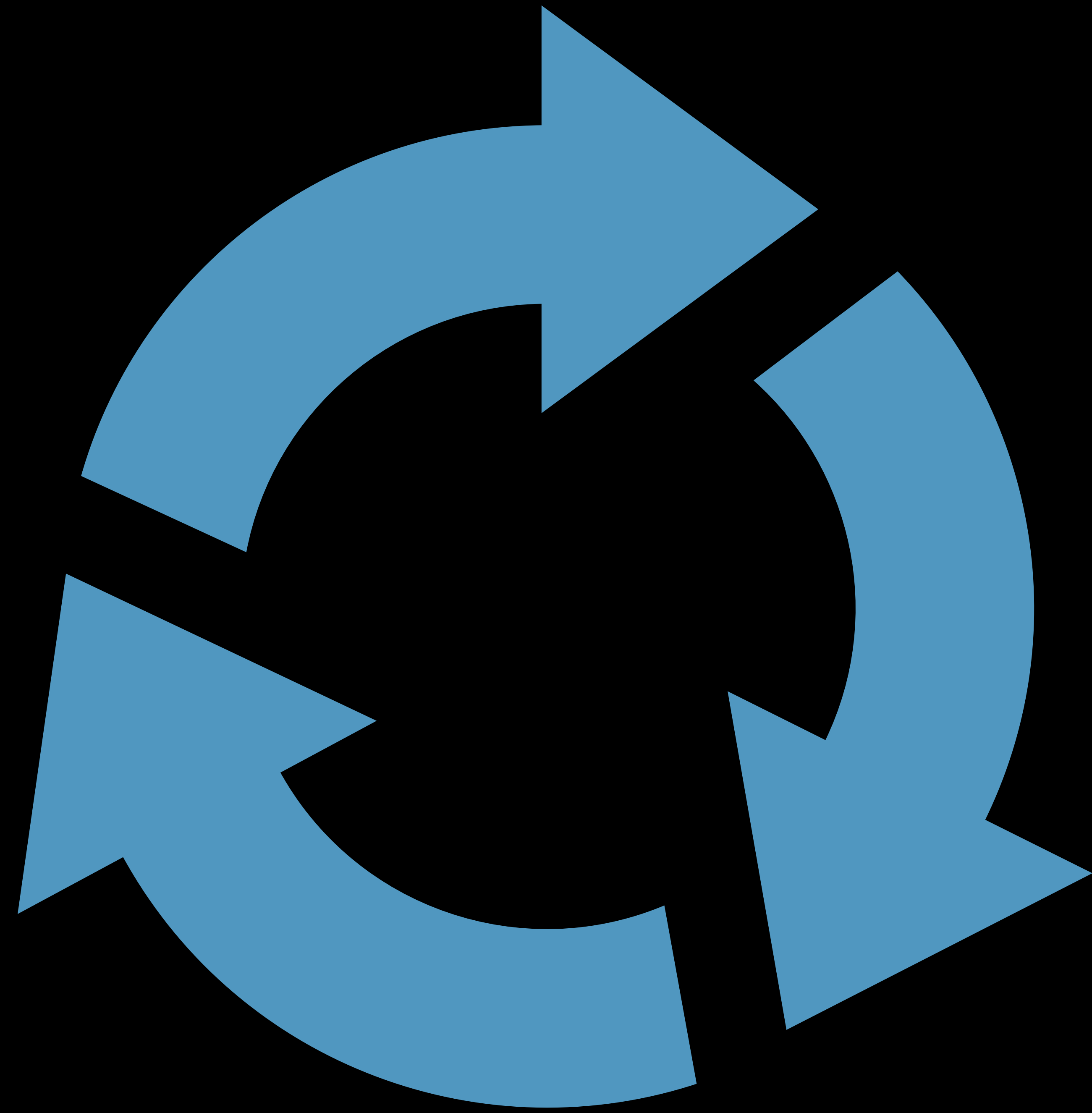
Manjeet Chawla, App Store Commerce

Subscription Lifecycle



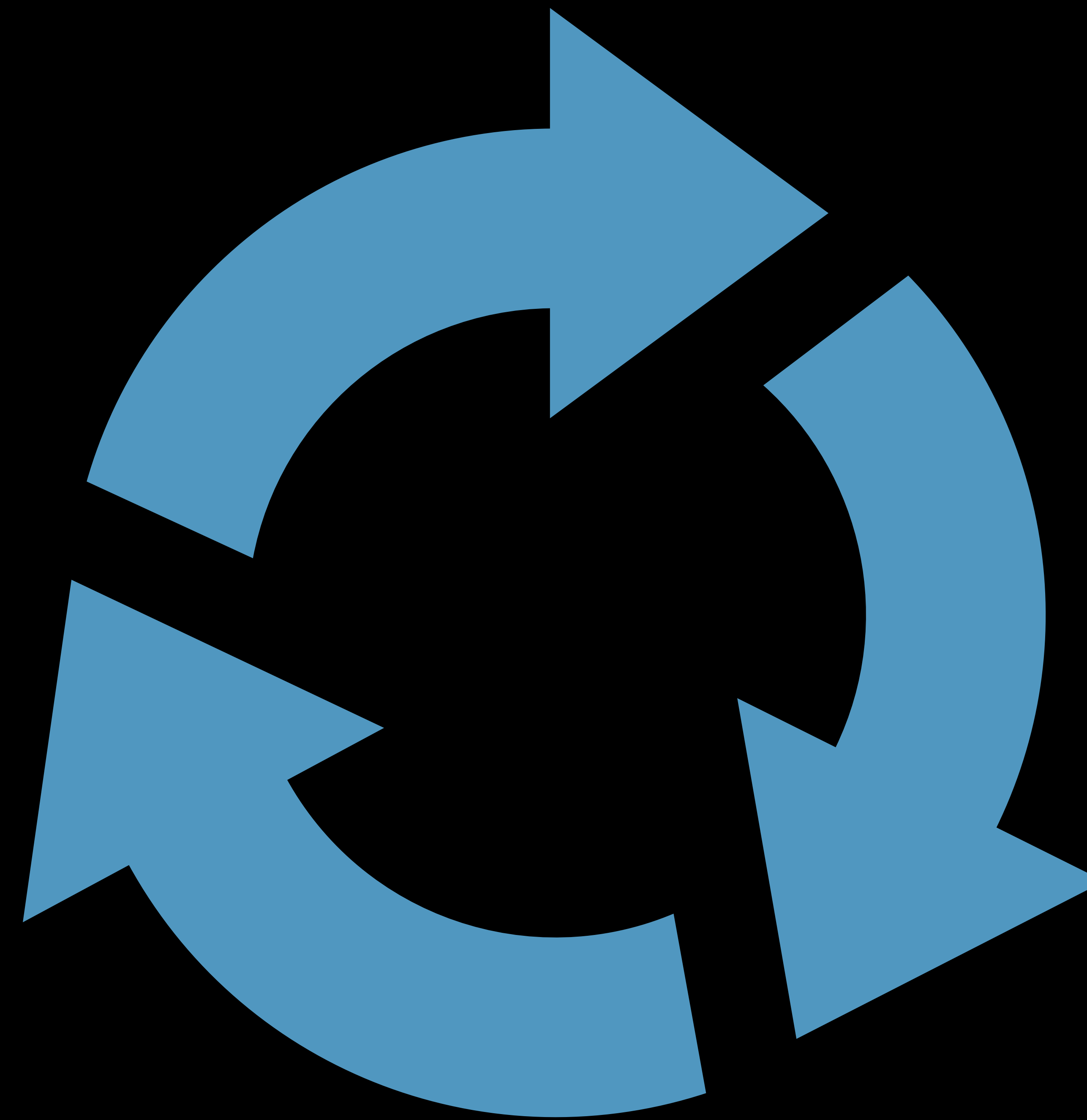
Subscription Lifecycle

Acquisition



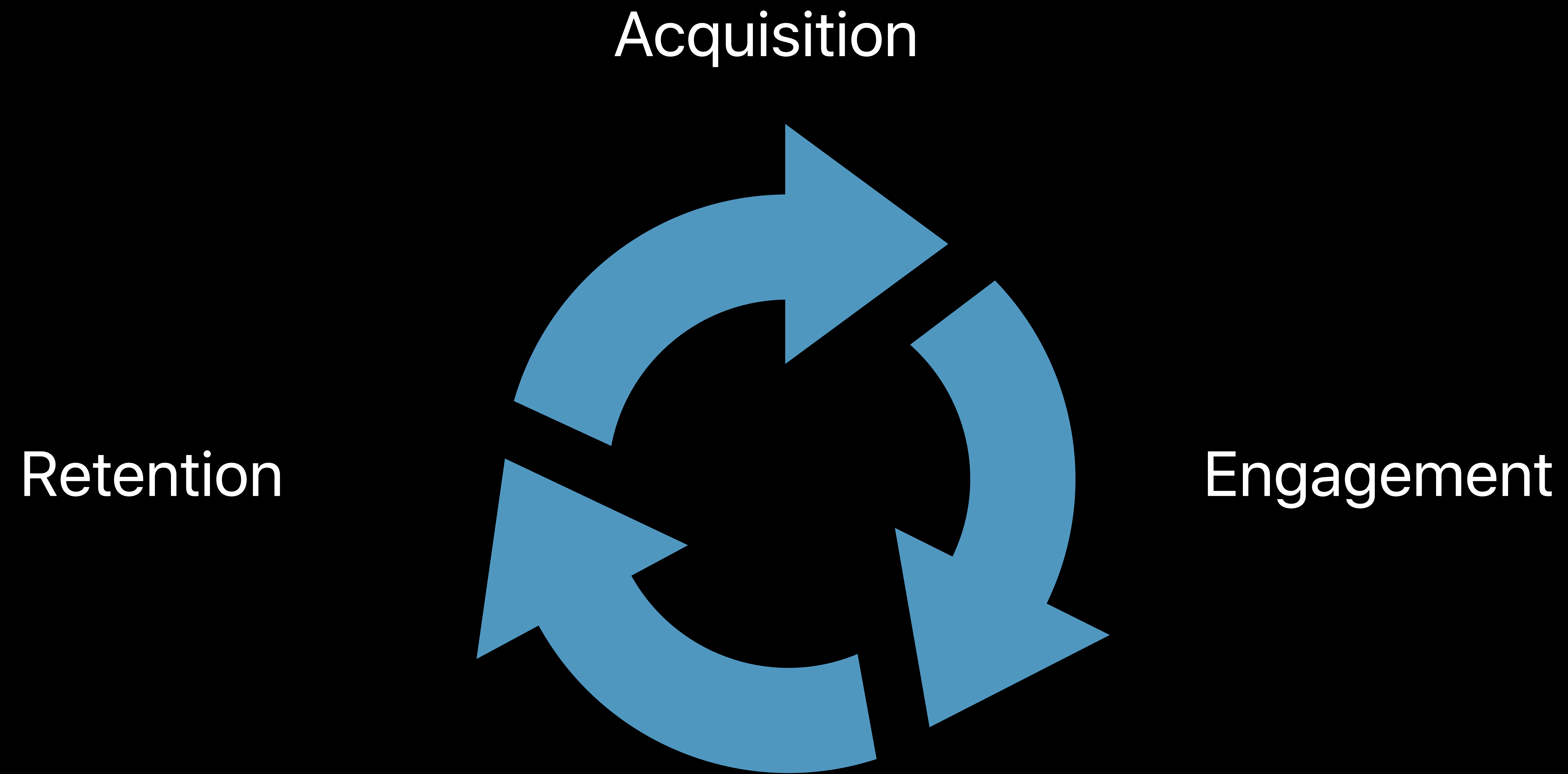
Subscription Lifecycle

Acquisition



Engagement

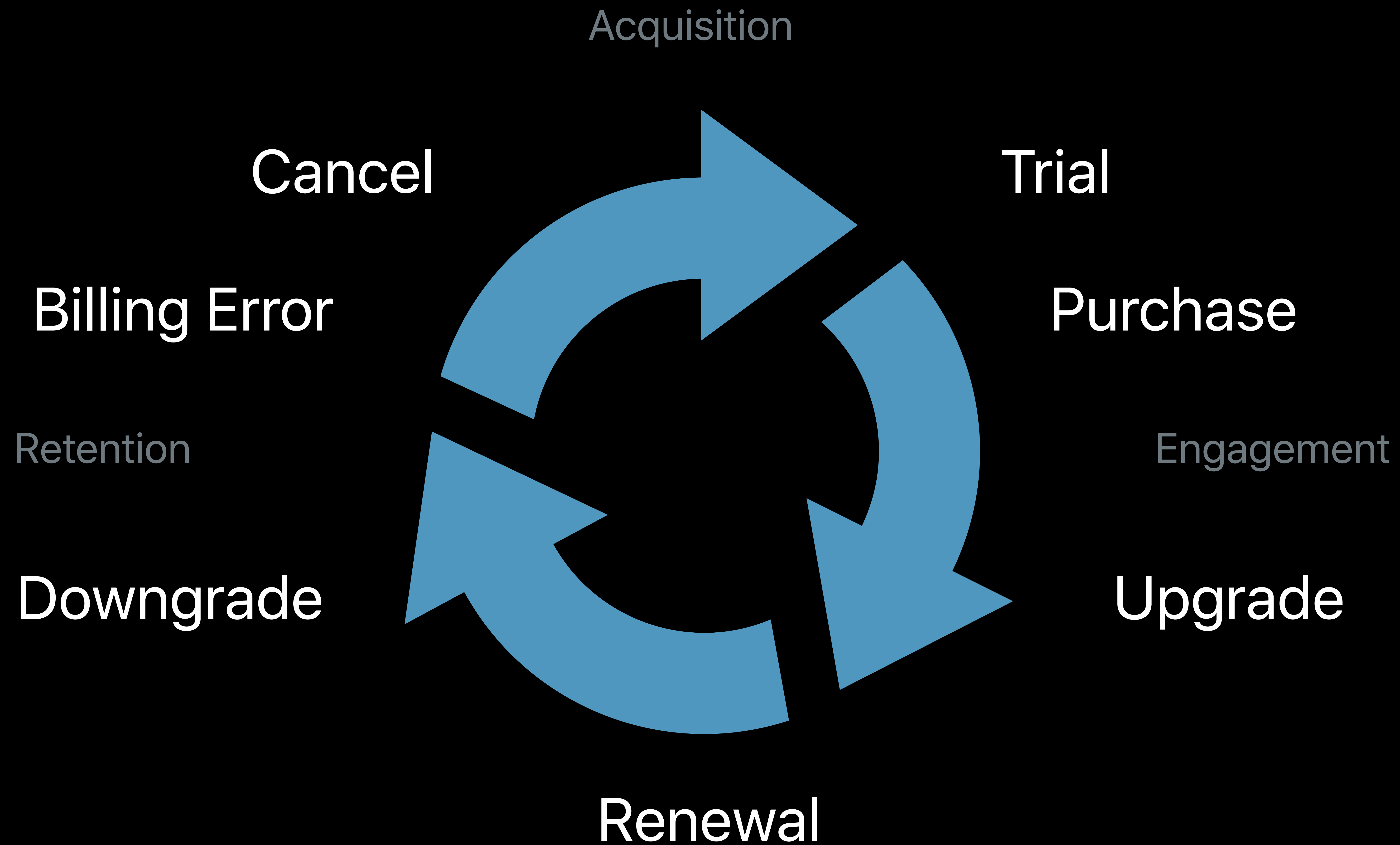
Subscription Lifecycle



Subscription Billing Events



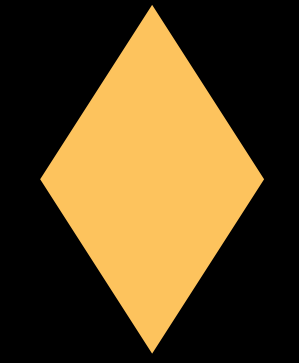
Subscription Billing Events



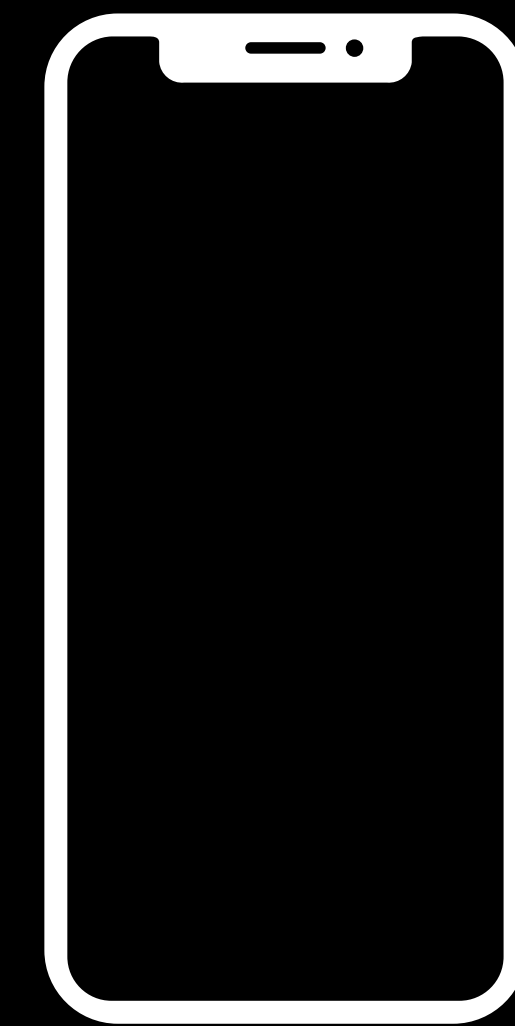
Purchase



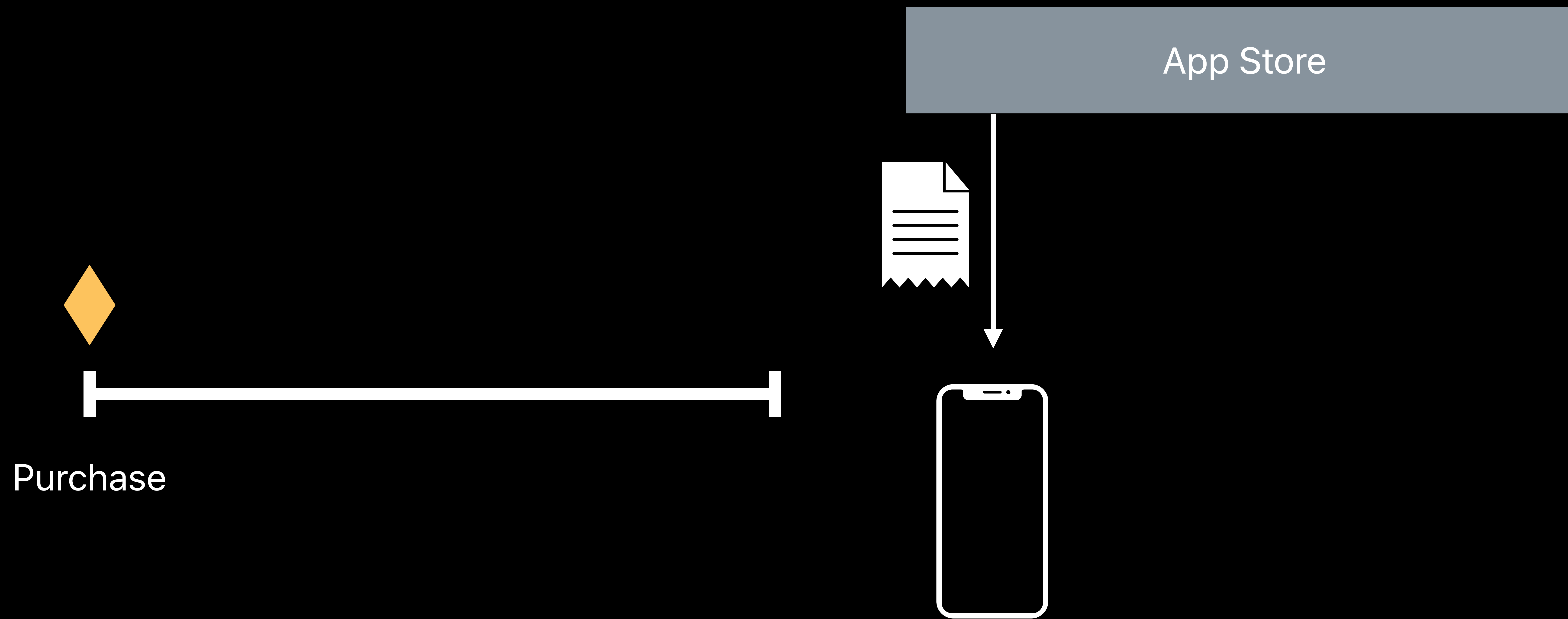
Purchase



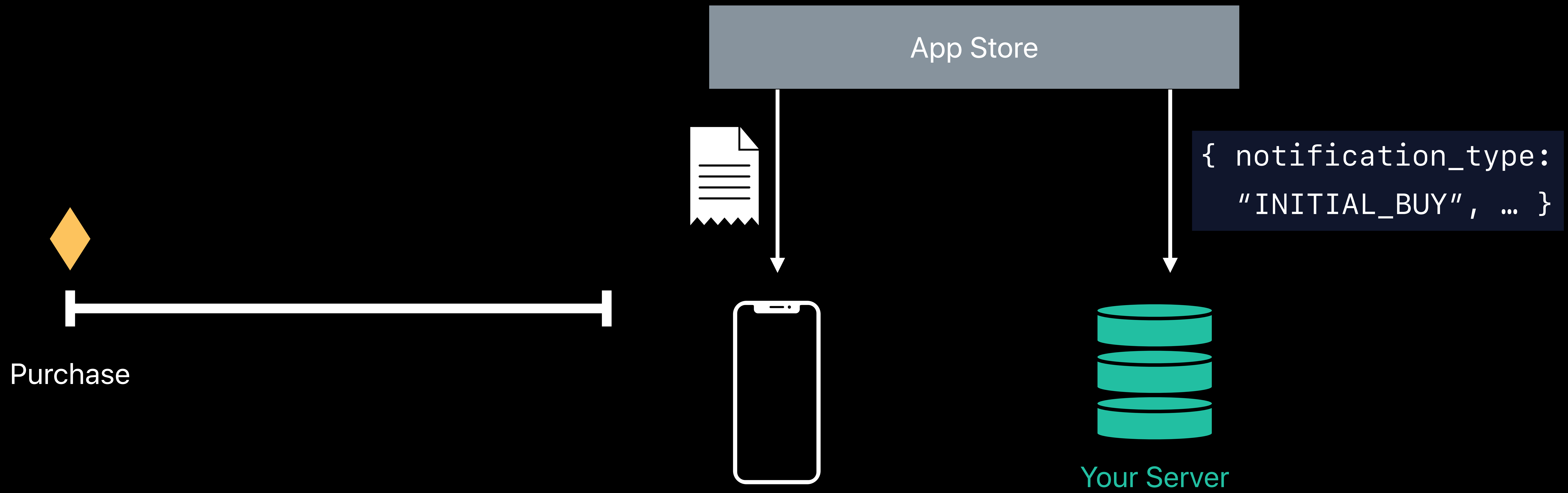
Purchase



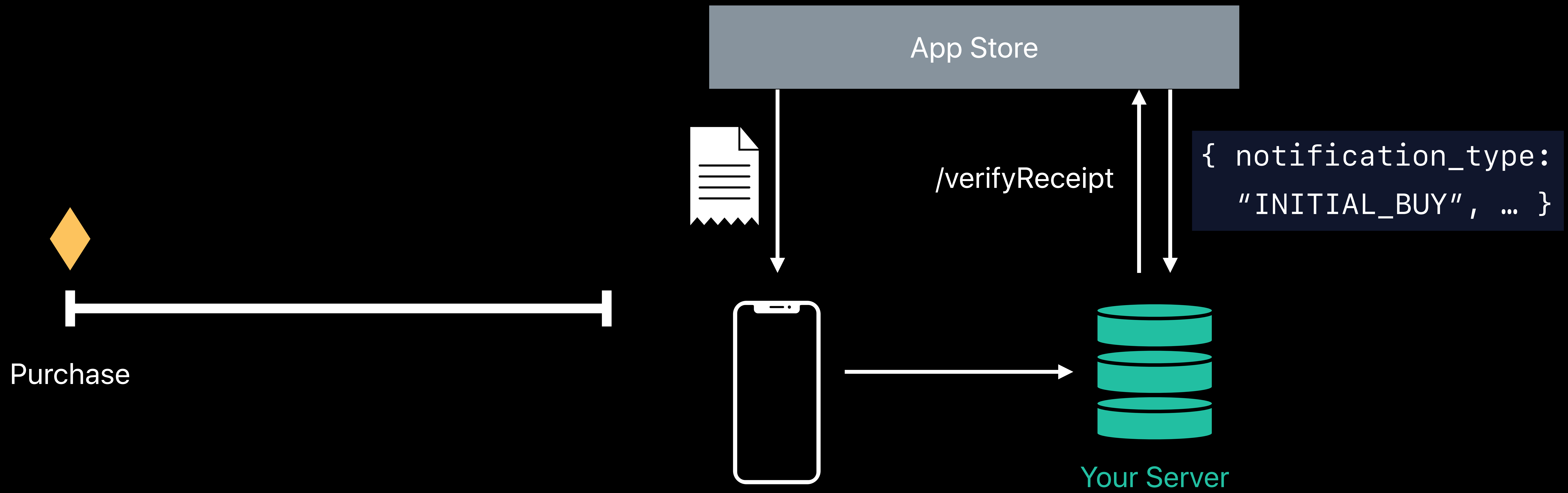
Purchase



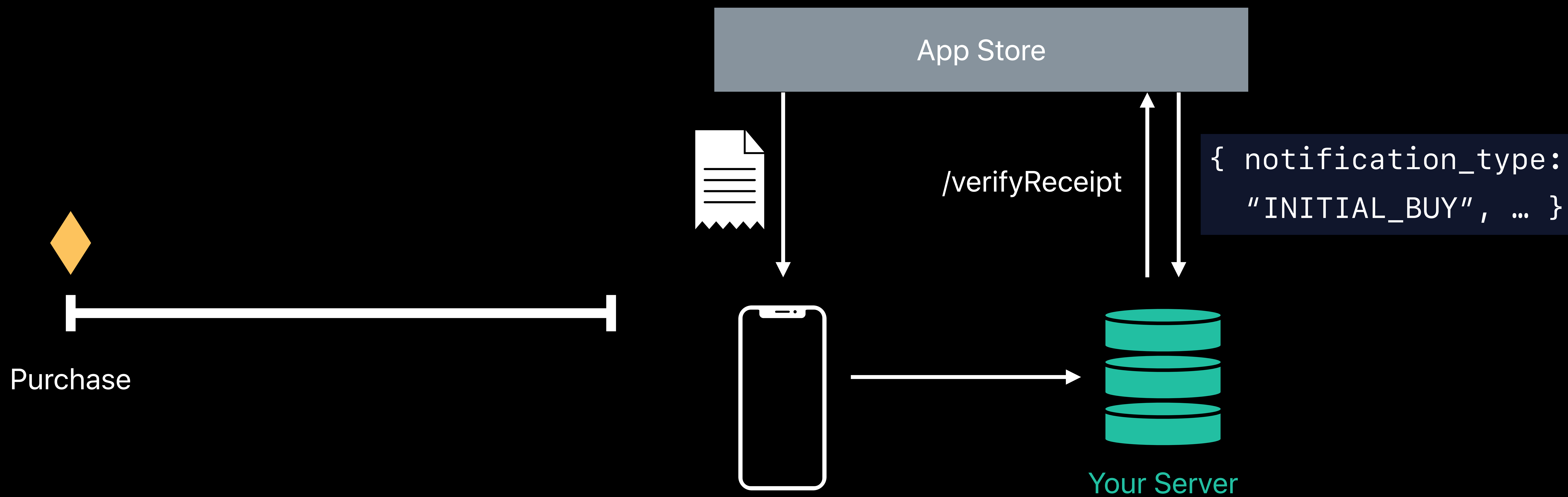
Purchase



Purchase

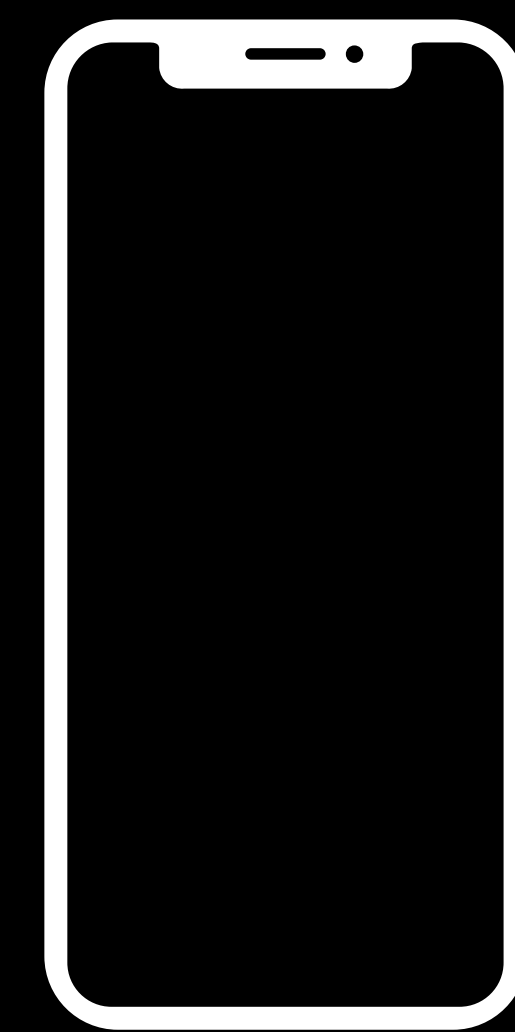
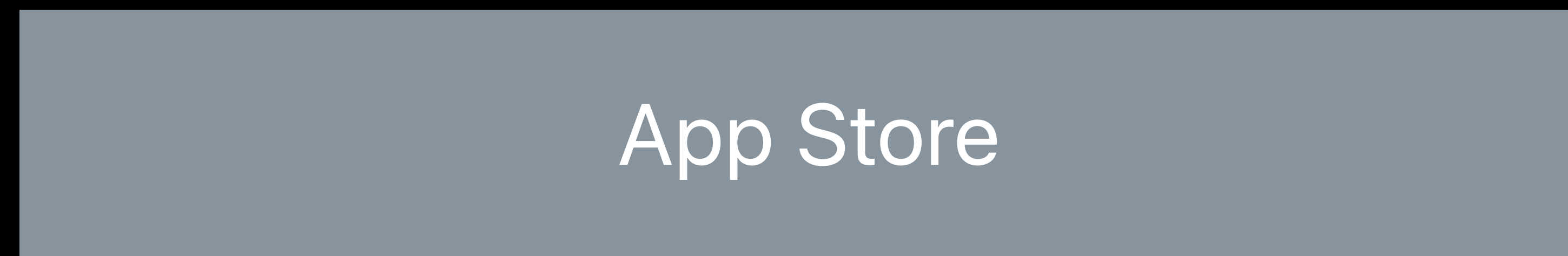


Purchase



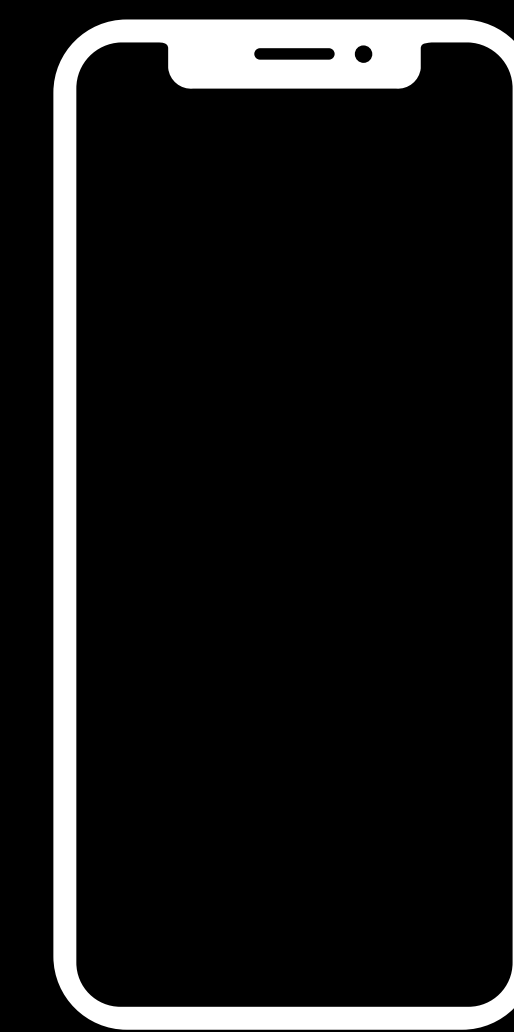
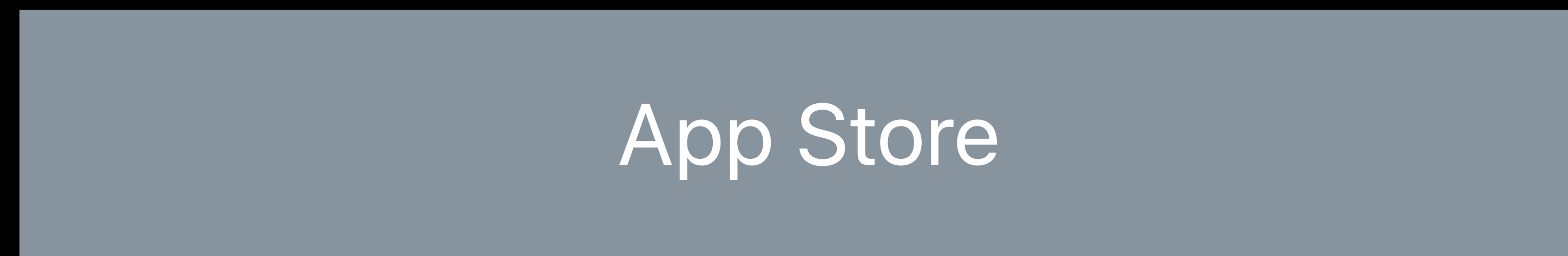
userId	originalTransactionId	latestExpiresDate	latestReceiptData	isSubscribed	willAutoRenew	webOrderLineItemId
90000001	1133557799	2019-06-01...	d24Fs...kJ87dDGe3=	TRUE	TRUE	123456789

Renewal



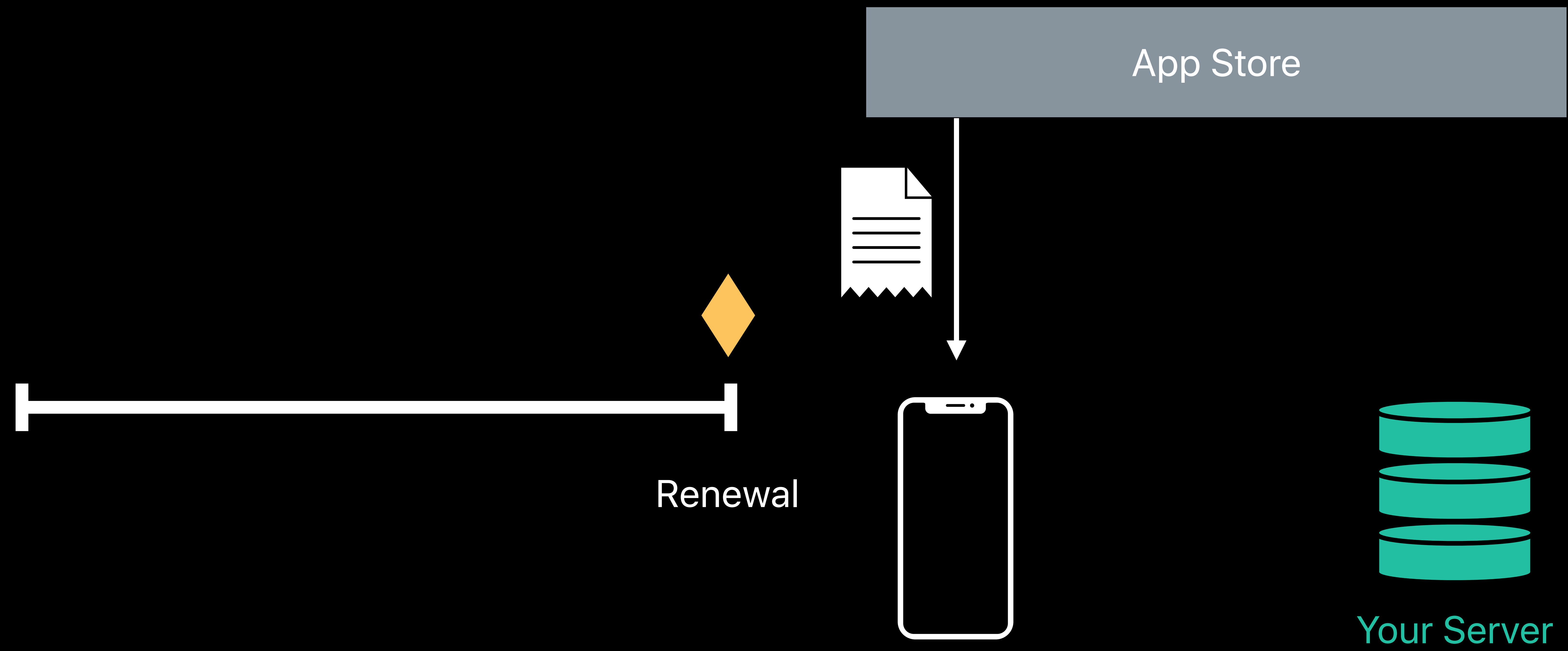
Your Server

Renewal

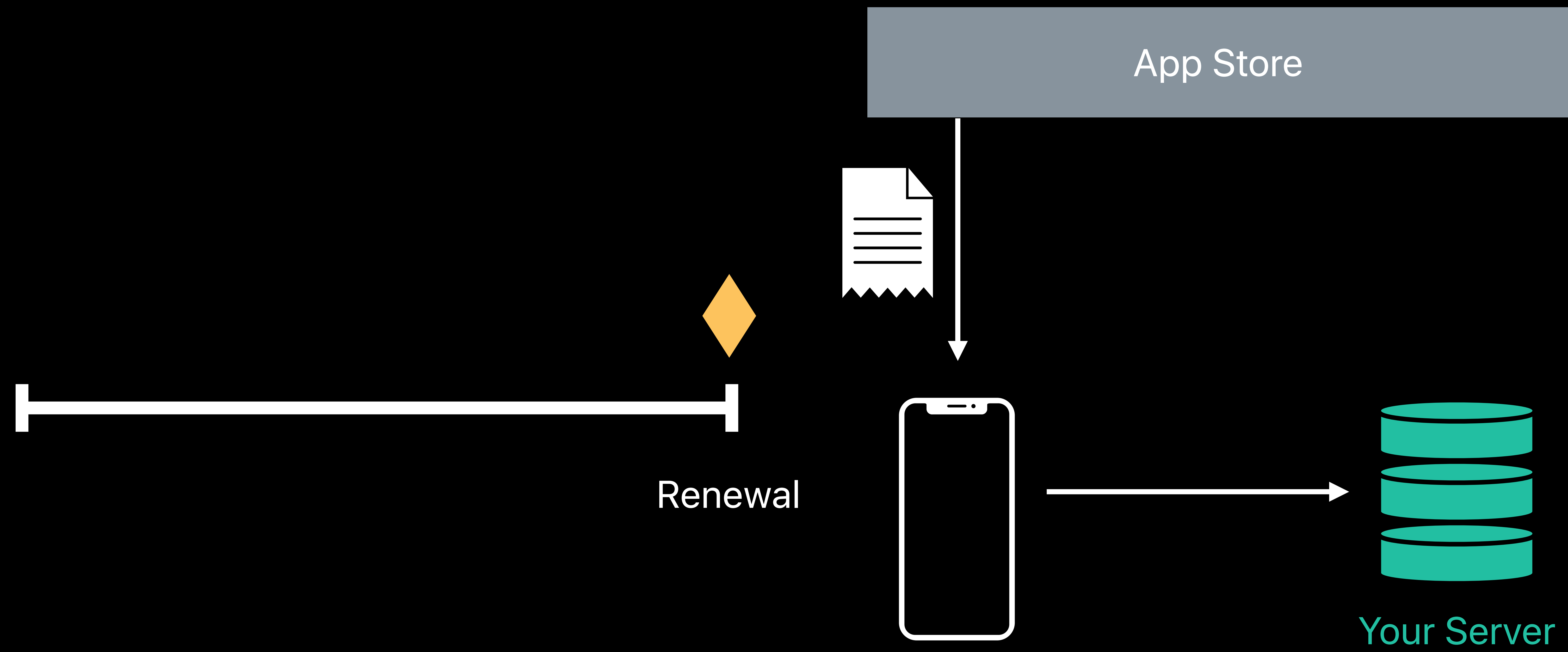


Your Server

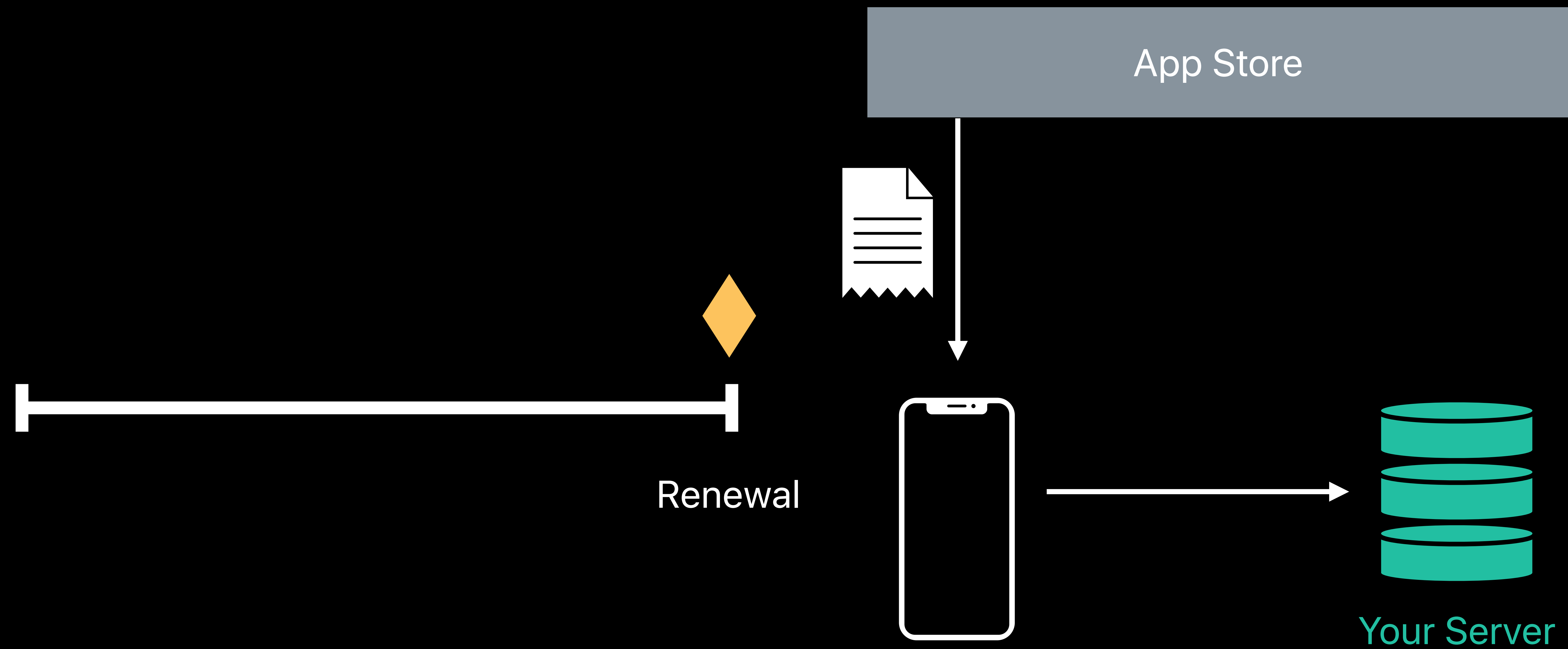
Renewal



Renewal

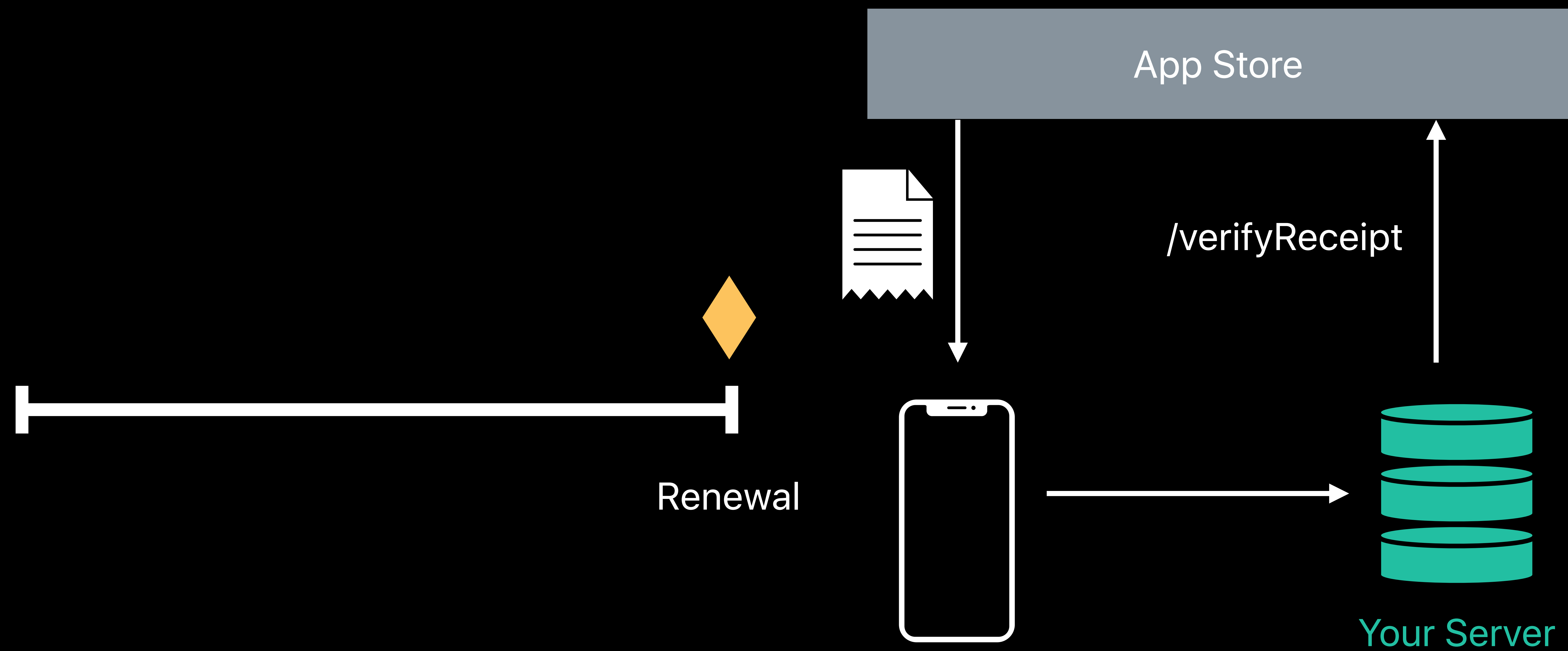


Renewal



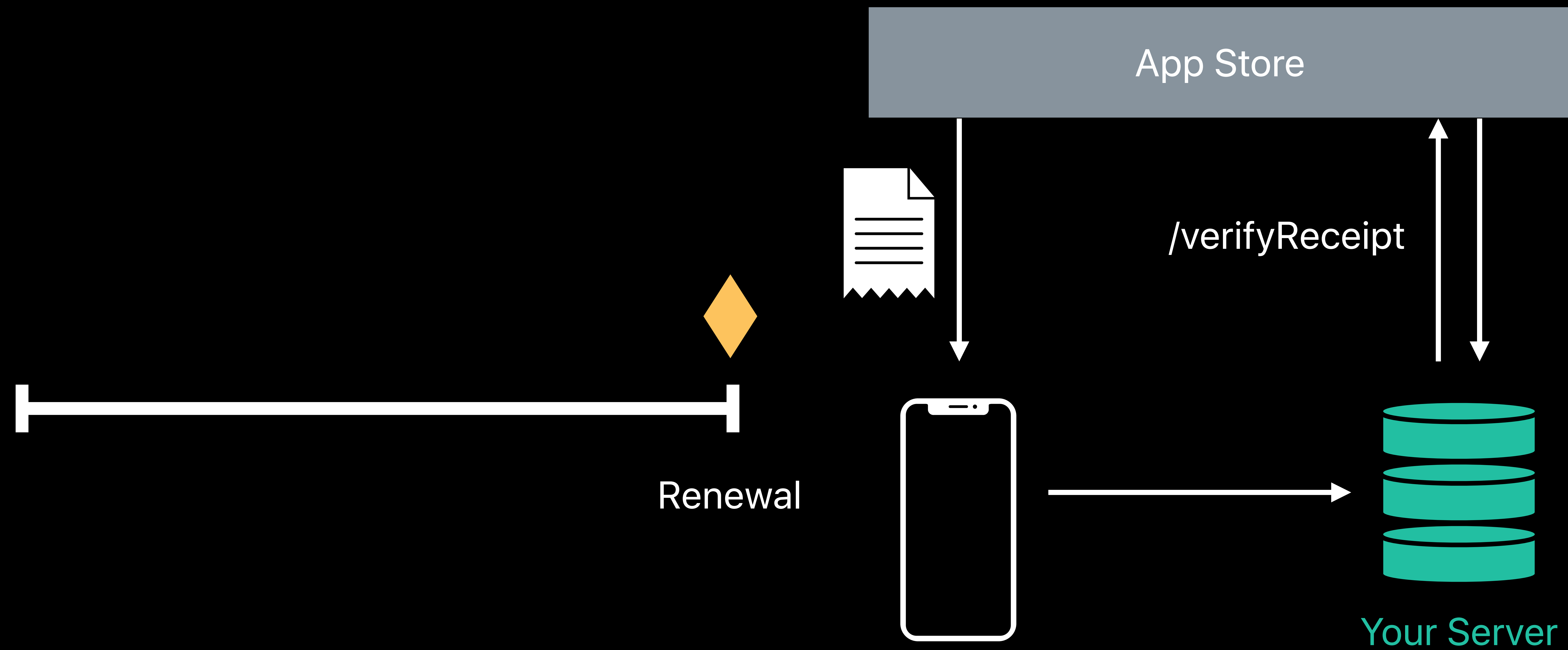
userId	originalTransactionId	latestExpiresDate	latestReceiptData	isSubscribed	willAutoRenew	webOrderLineItemId
90000001	1133557799	2019-06-01...	d24Fs...kJ87dDGe3=	TRUE	TRUE	123456789

Renewal



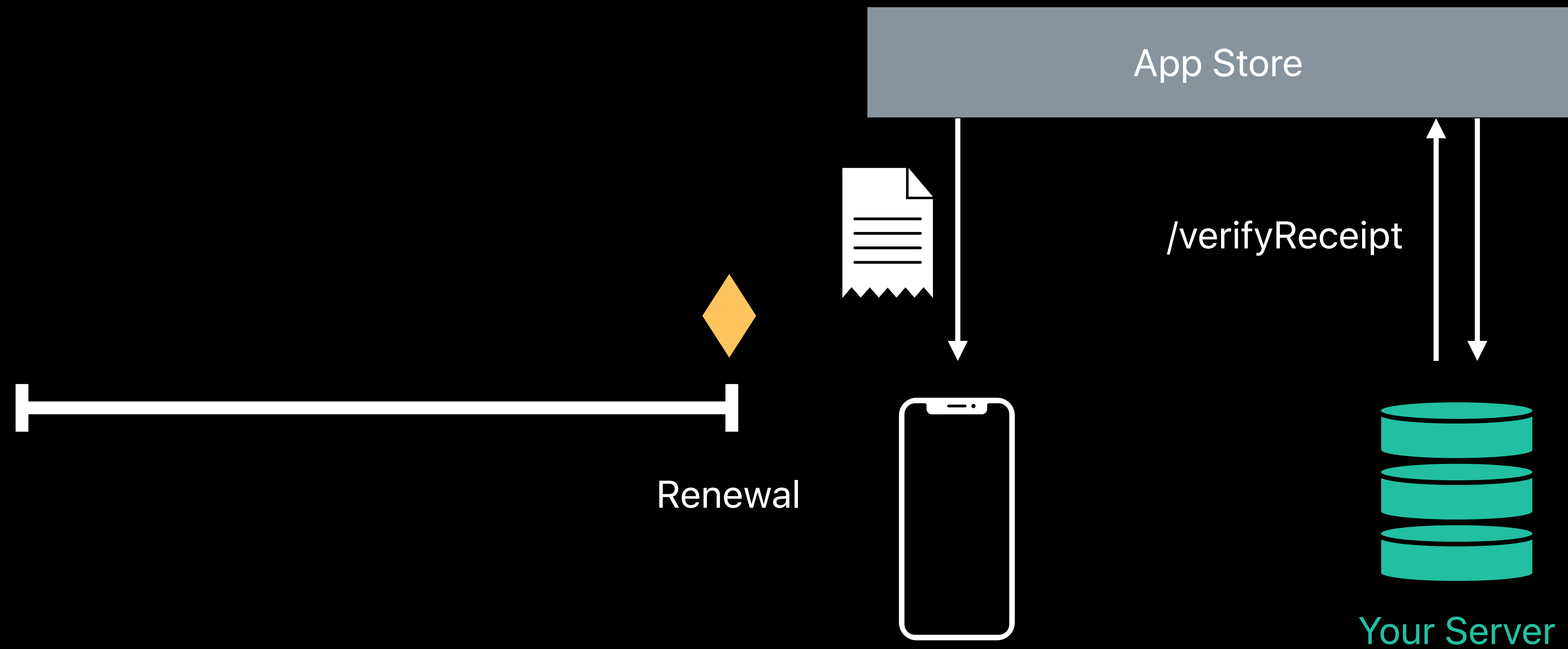
userId	originalTransactionId	latestExpiresDate	latestReceiptData	isSubscribed	willAutoRenew	webOrderLineItemId
90000001	1133557799	2019-06-01...	d24Fs...kJ87dDGe3=	TRUE	TRUE	123456789

Renewal



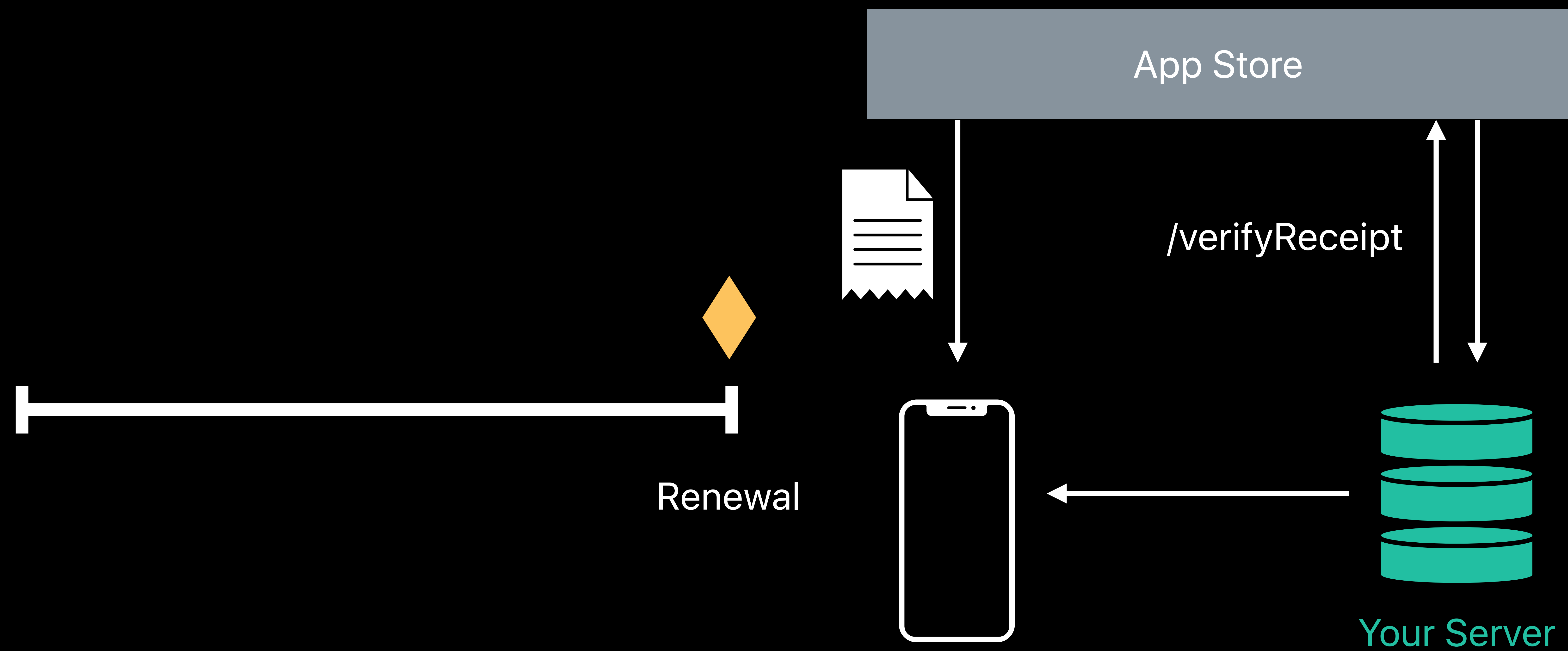
userId	originalTransactionId	latestExpiresDate	latestReceiptData	isSubscribed	willAutoRenew	webOrderLineItemId
90000001	1133557799	2019-06-01...	d24Fs...kJ87dDGe3=	TRUE	TRUE	123456789

Renewal



userId	originalTransactionId	latestExpiresDate	latestReceiptData	isSubscribed	willAutoRenew	webOrderLineItemId
90000001	1133557799	2019-07-01...	d24Fs...kJ87dDGe3=	TRUE	TRUE	123456790

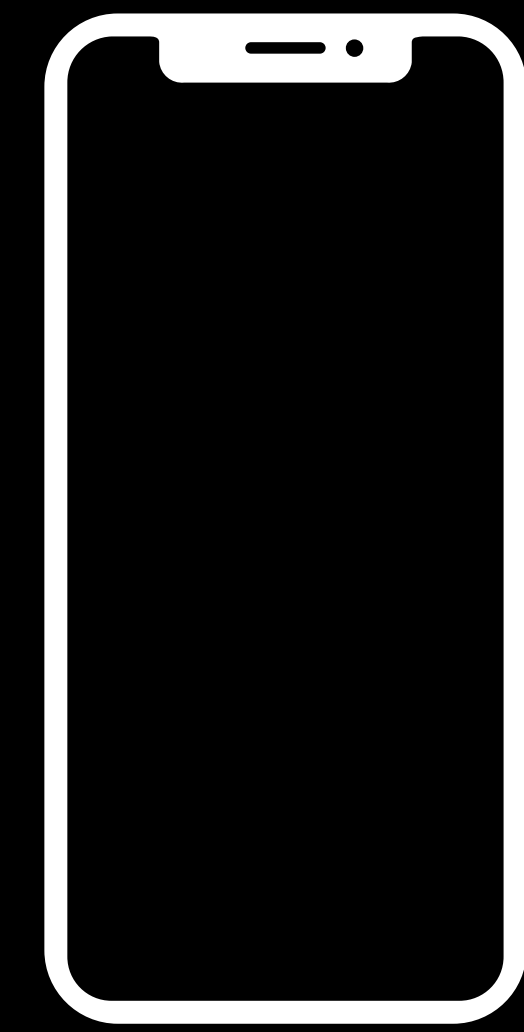
Renewal



userId	originalTransactionId	latestExpiresDate	latestReceiptData	isSubscribed	willAutoRenew	webOrderLineItemId
90000001	1133557799	2019-07-01...	d24Fs...kJ87dDGe3=	TRUE	TRUE	123456790

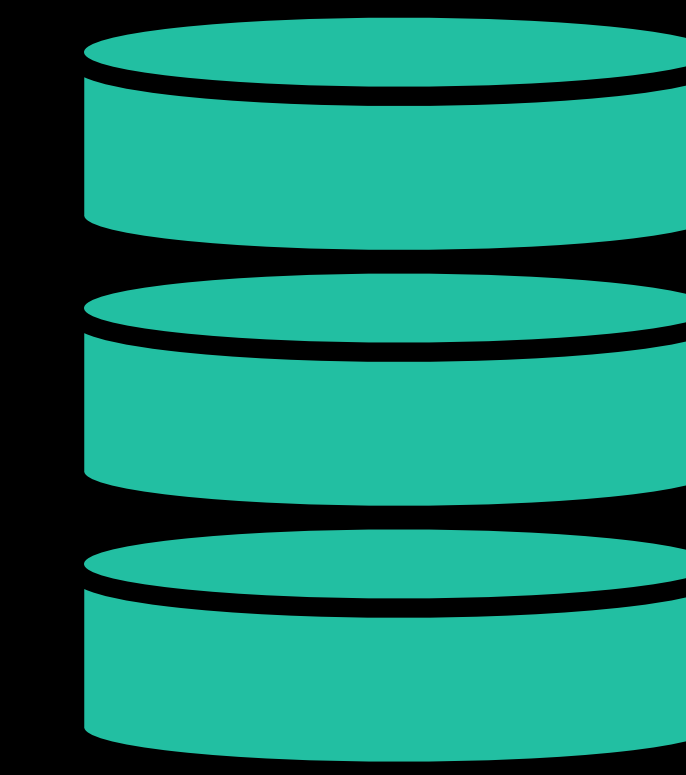
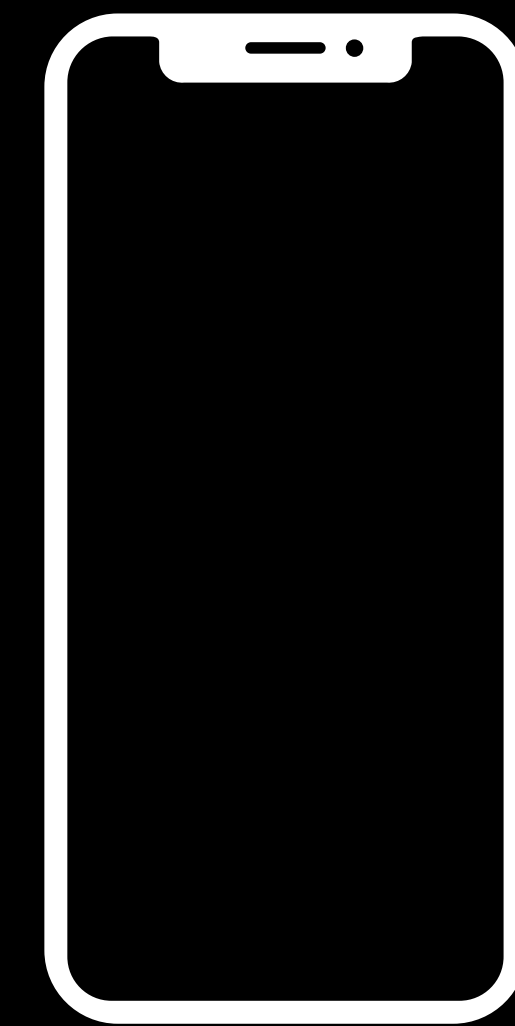
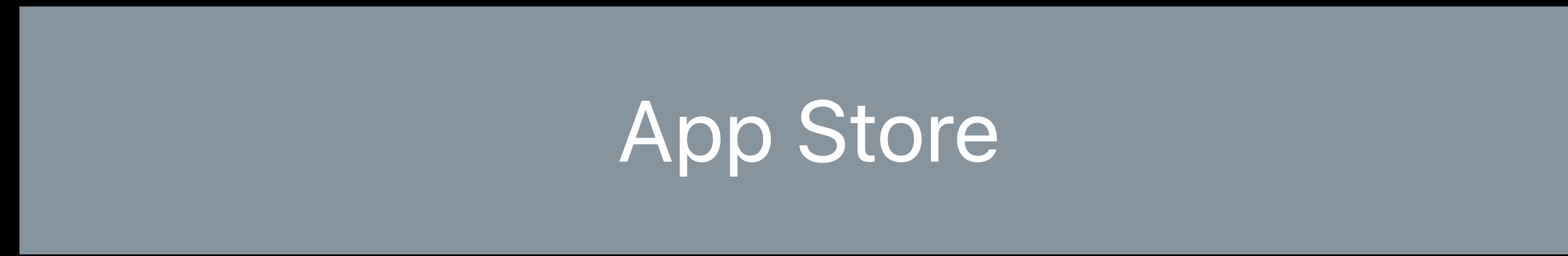
Upgrade

App Store



Your Server

Upgrade



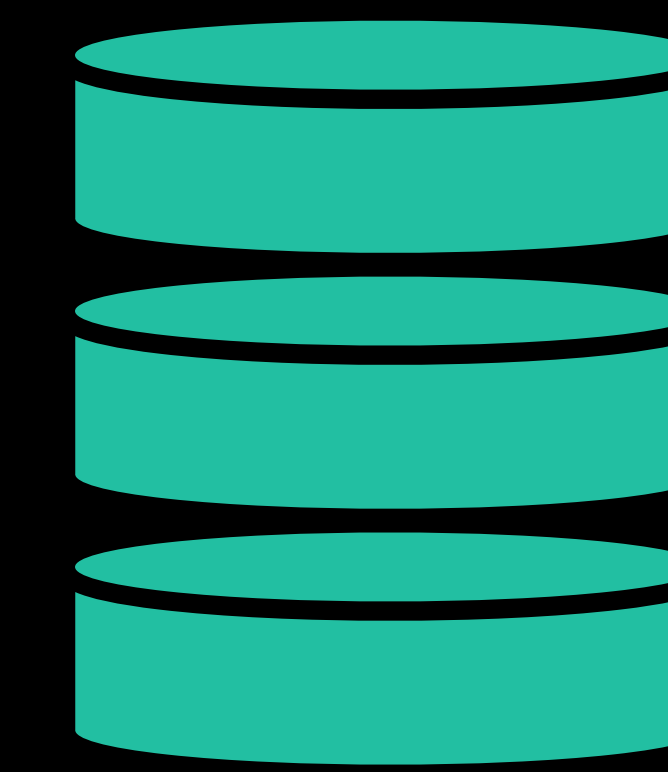
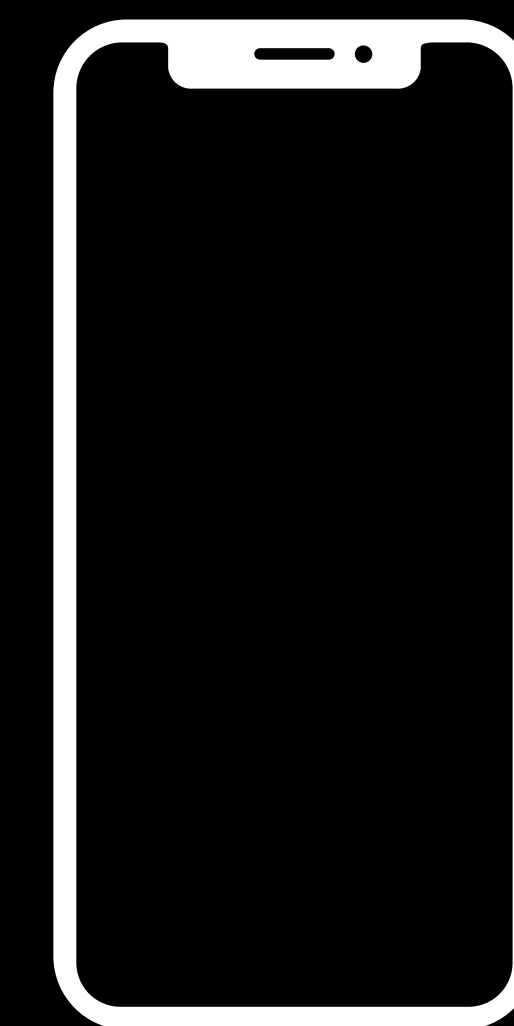
Your Server

Upgrade



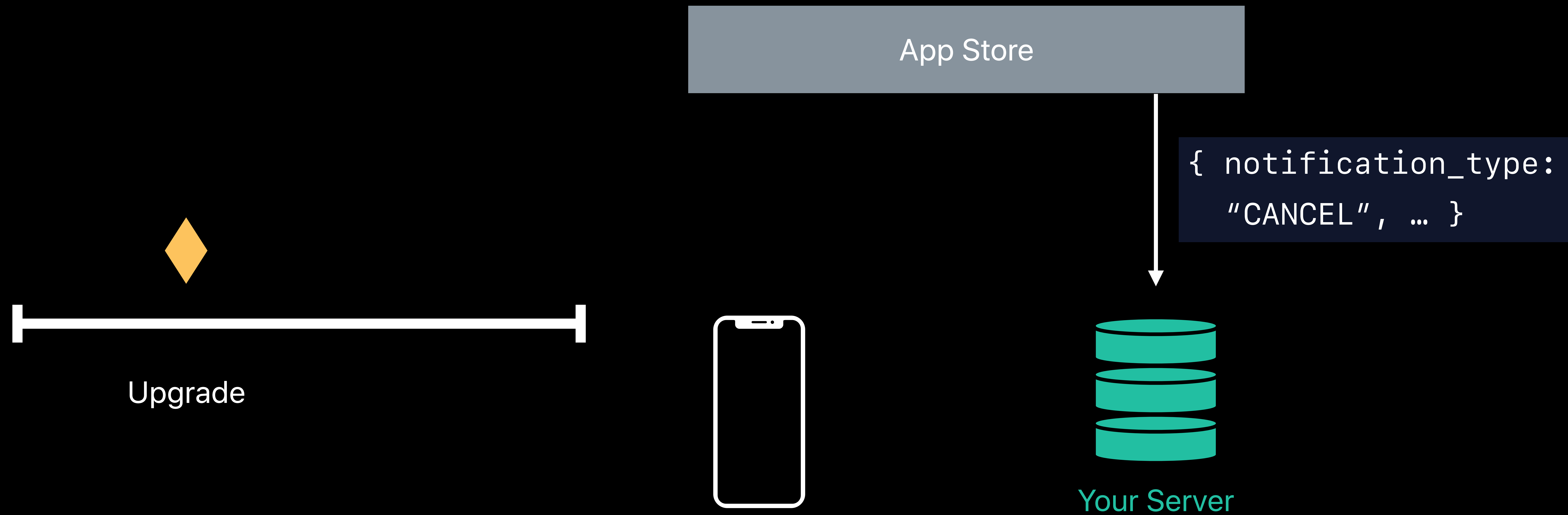
App Store

```
{ notification_type:  
  "CANCEL", ... }
```



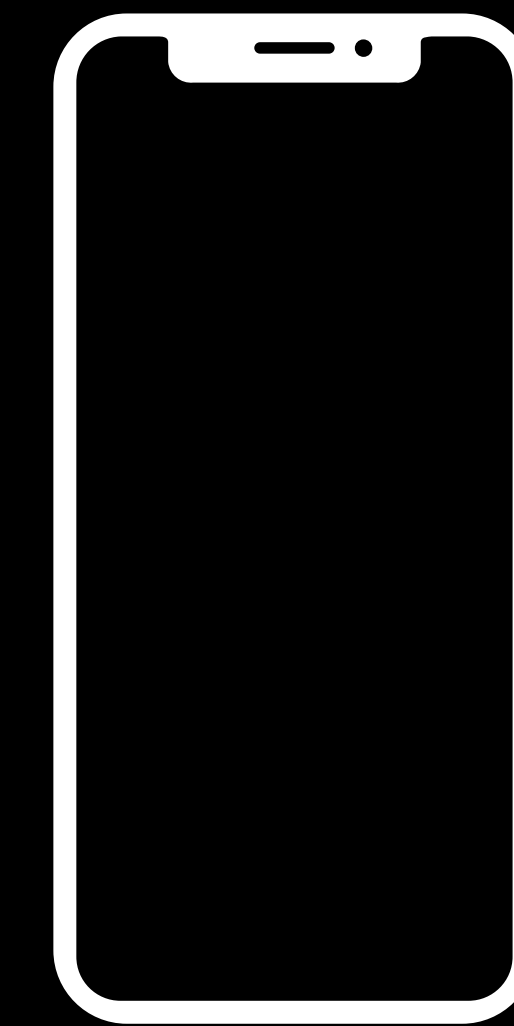
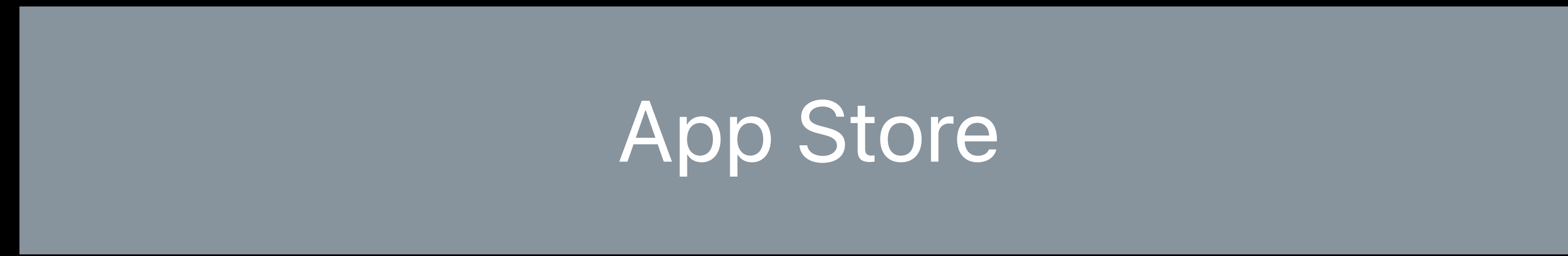
Your Server

Upgrade



userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	cancellationDate	subscriptionProductID
90000001	1133557799	2019-07-01...	TRUE	TRUE	2019-07-15...	com.product.basic

Upgrade



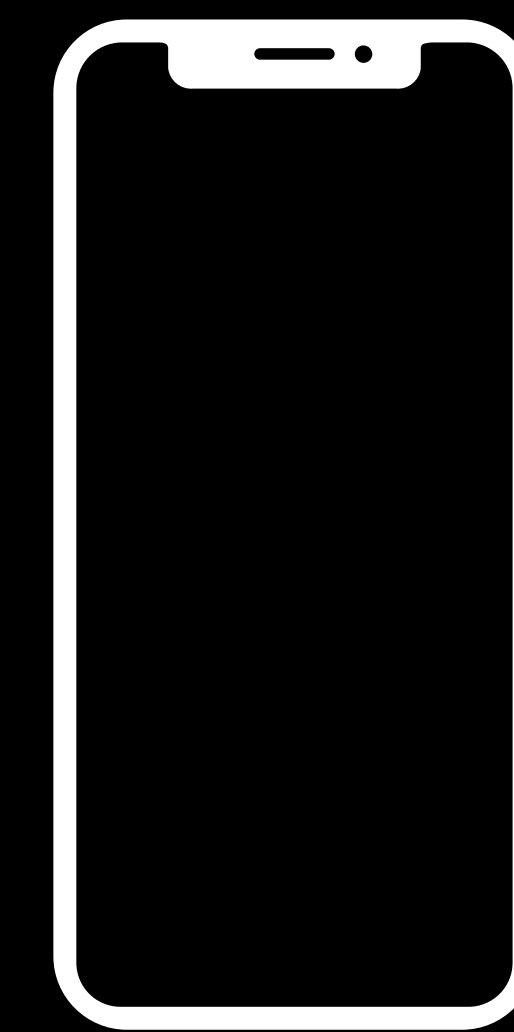
Your Server

Upgrade



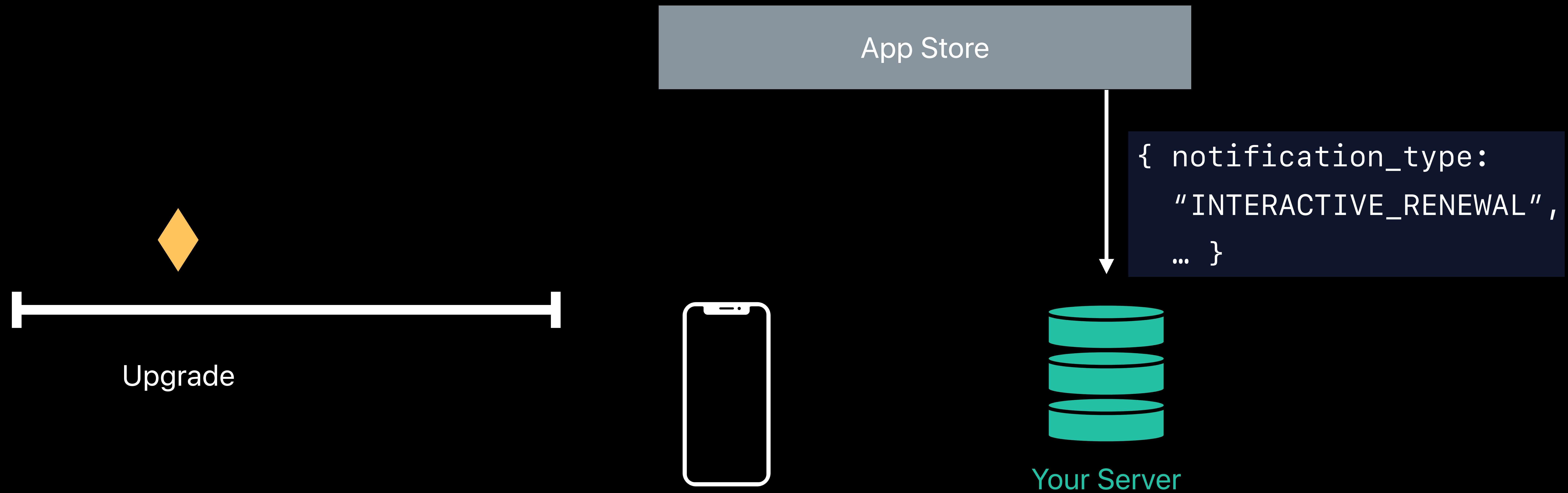
App Store

```
{ notification_type:  
  "INTERACTIVE_RENEWAL",  
  ... }  
}
```



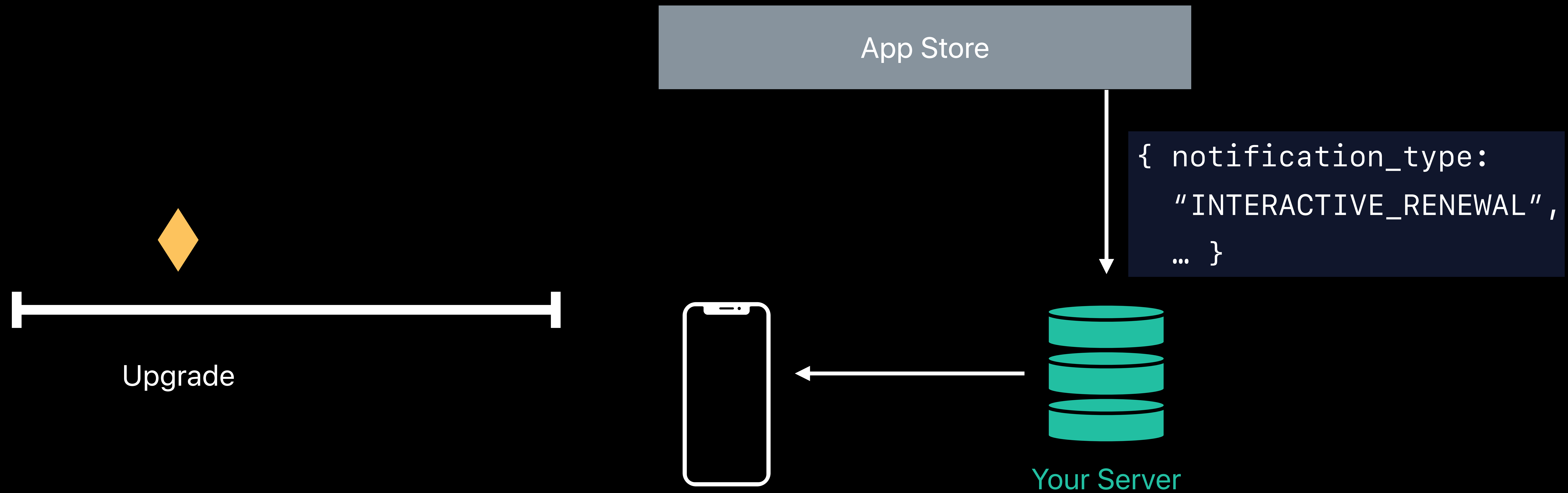
Your Server

Upgrade



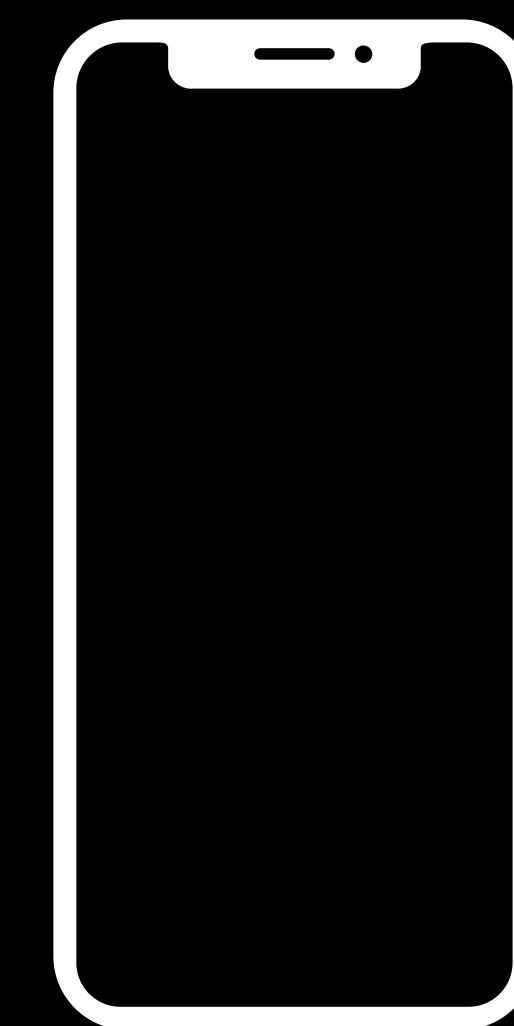
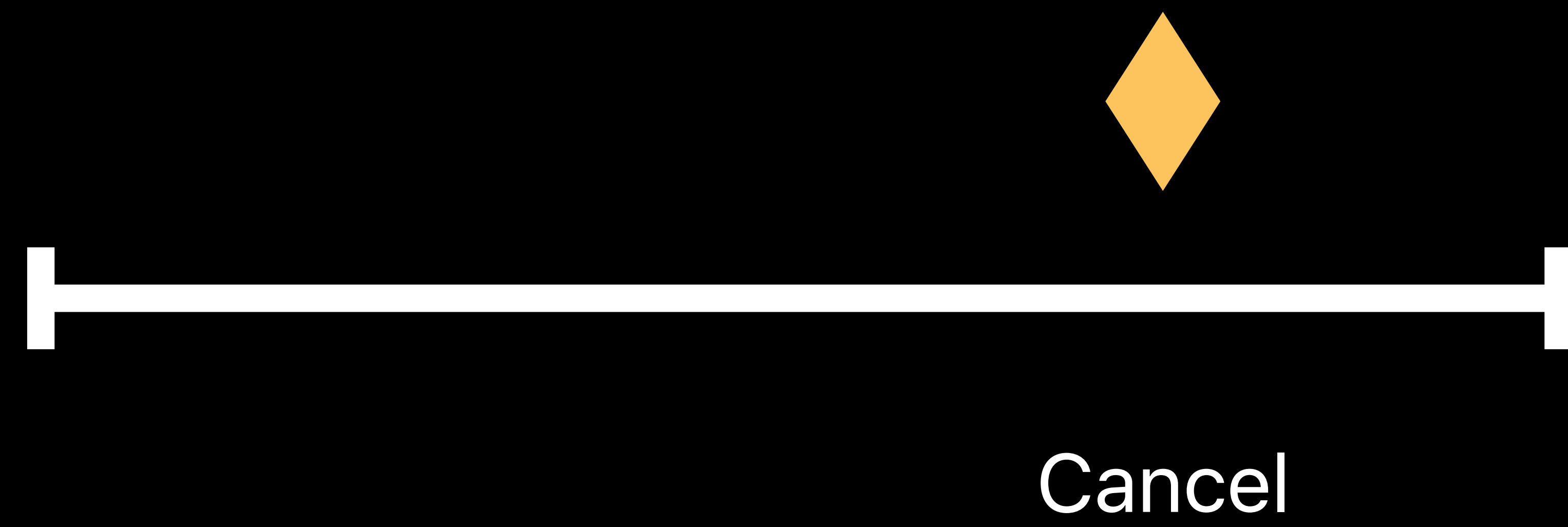
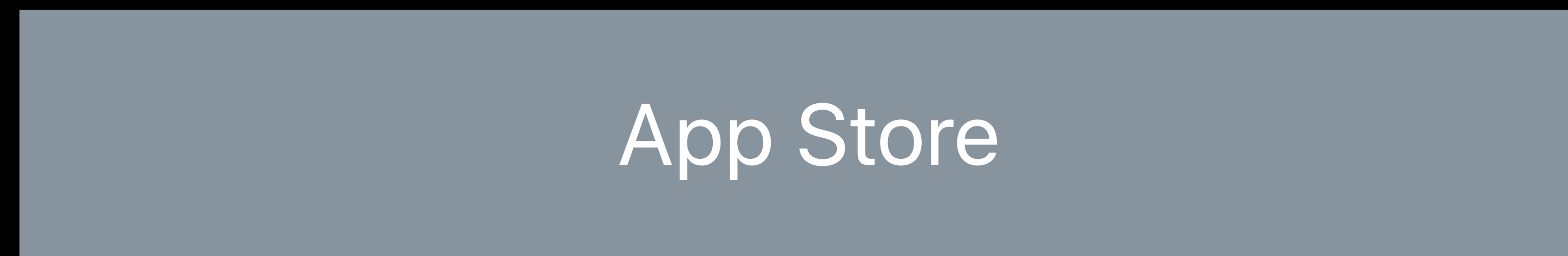
userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	upgradeDate	subscriptionProductID
90000001	1133557799	2019-07-01...	TRUE	TRUE	2019-07-15...	com.product.premium

Upgrade

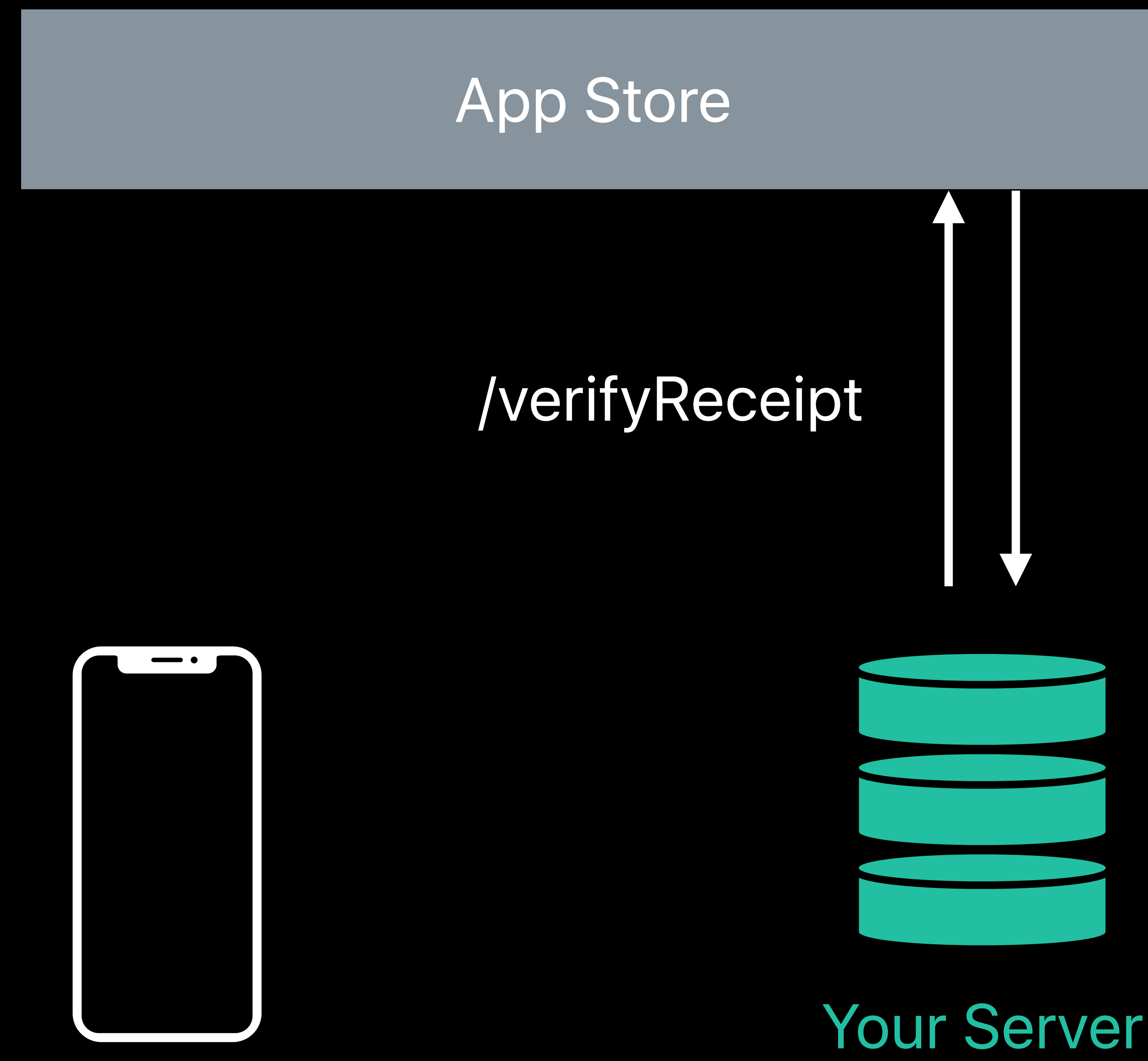
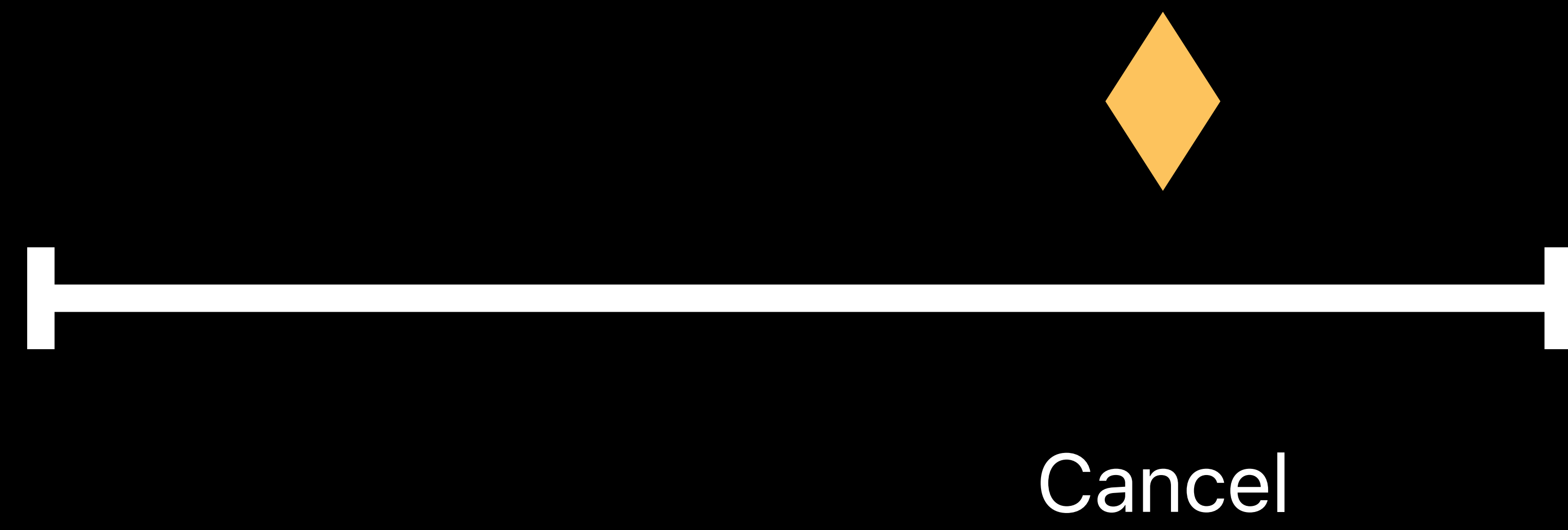


userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	upgradeDate	subscriptionProductID
90000001	1133557799	2019-07-01...	TRUE	TRUE	2019-07-15...	com.product.premium

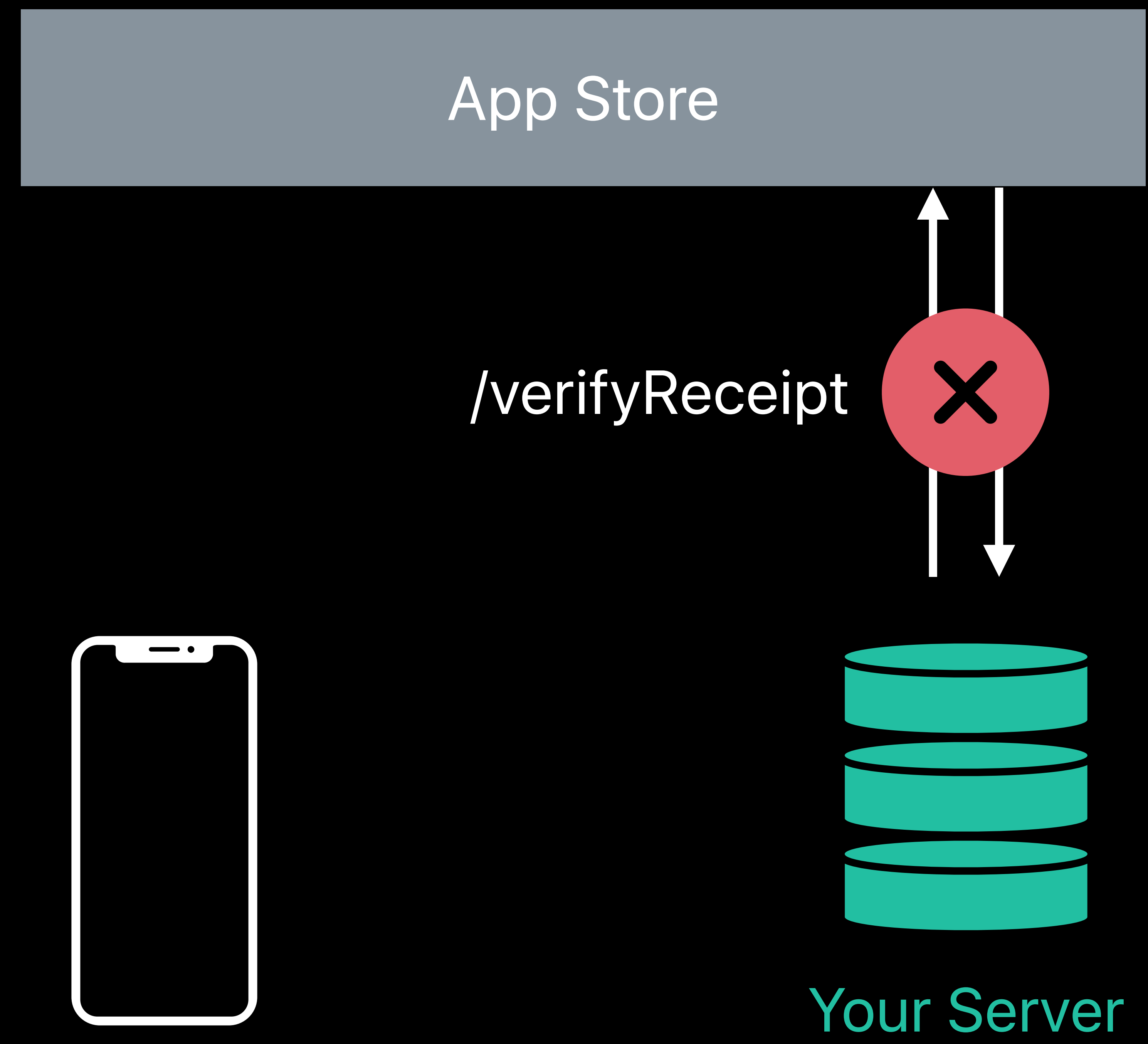
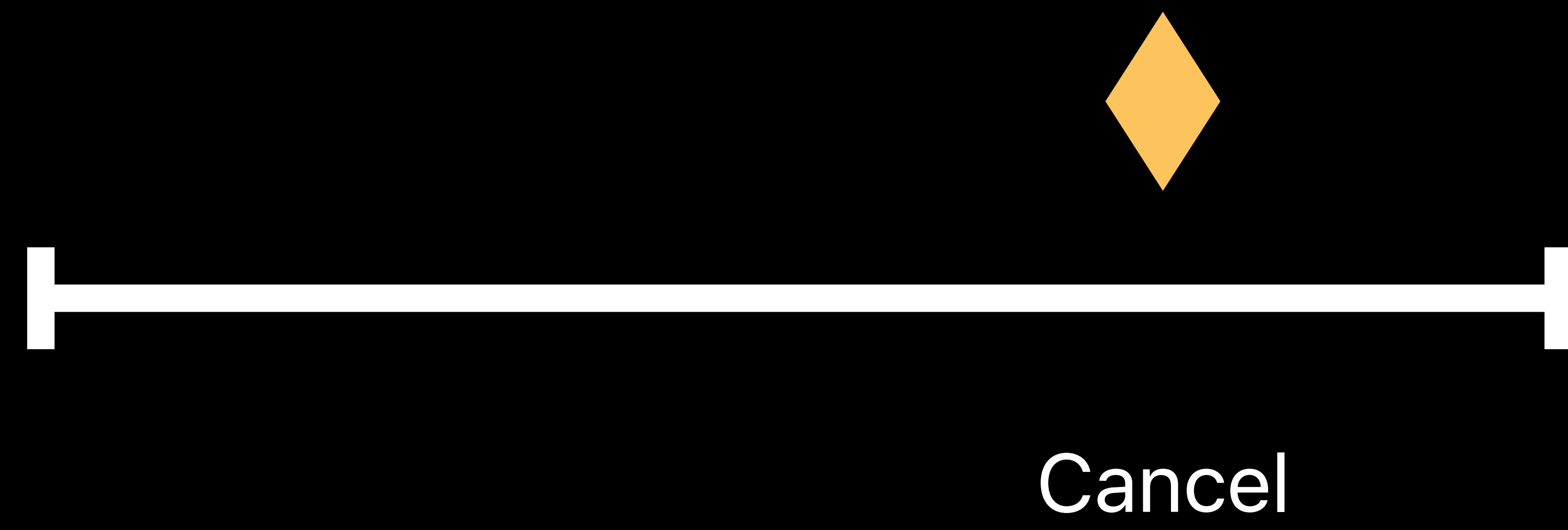
Cancel



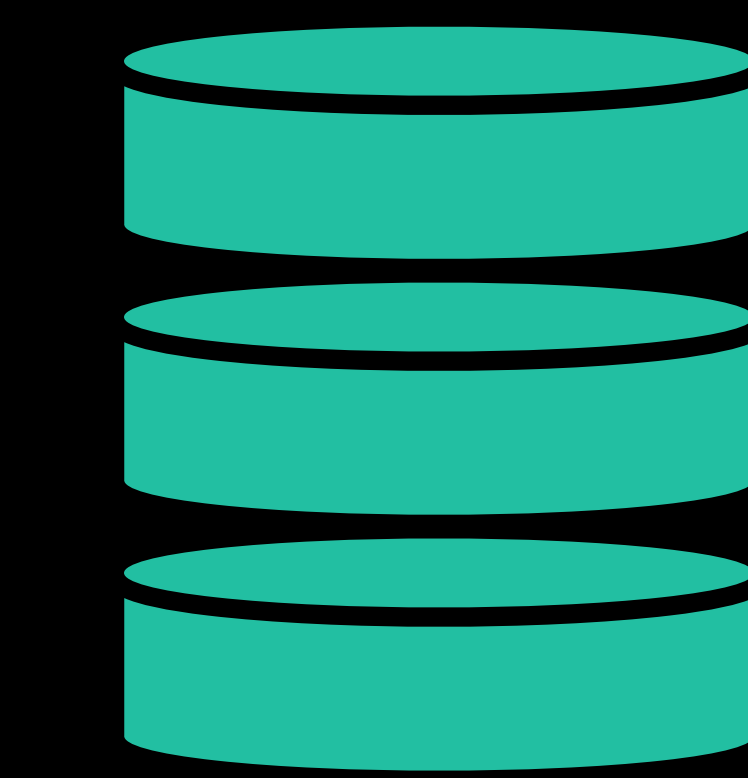
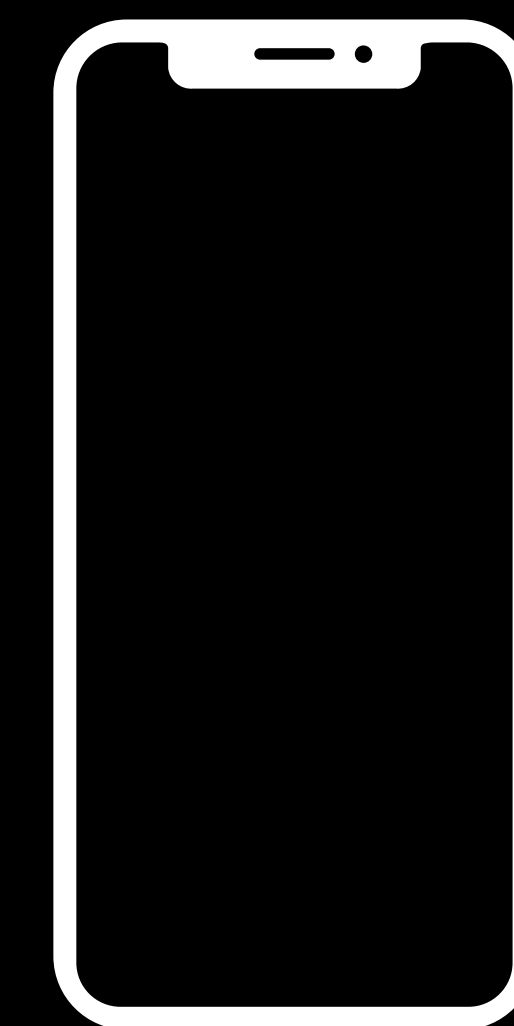
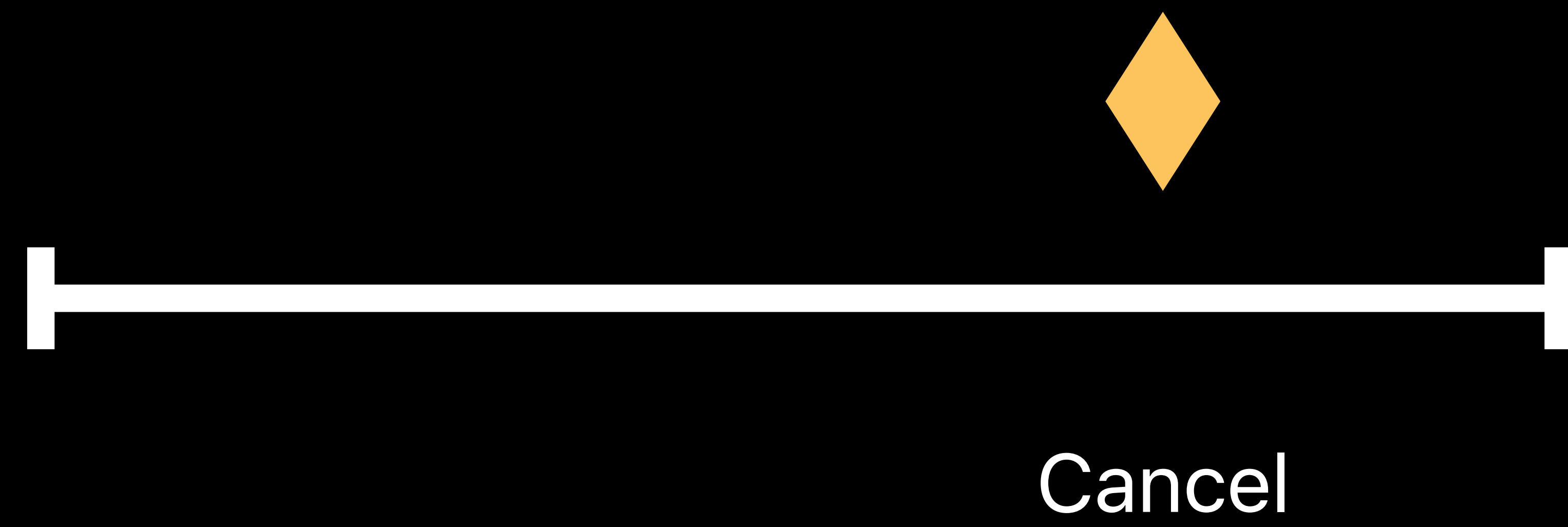
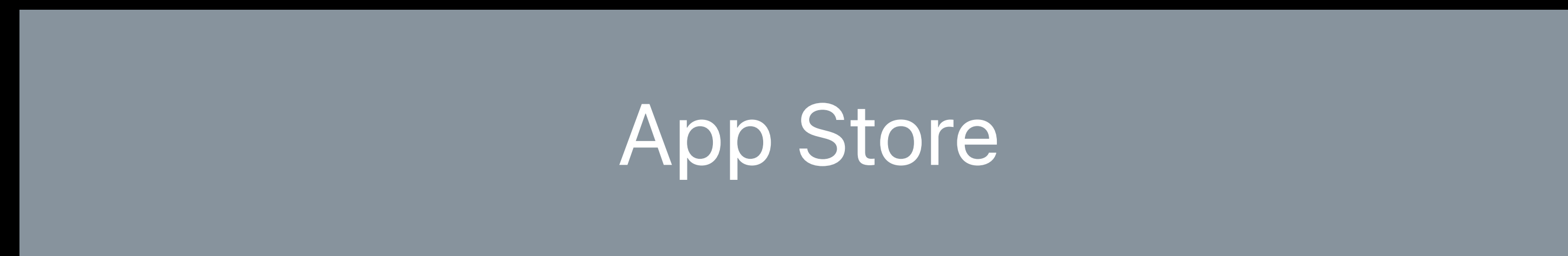
Cancel



Cancel



Cancel



Your Server

Cancel

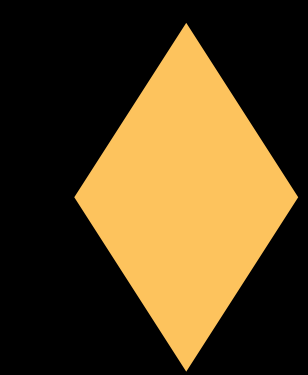
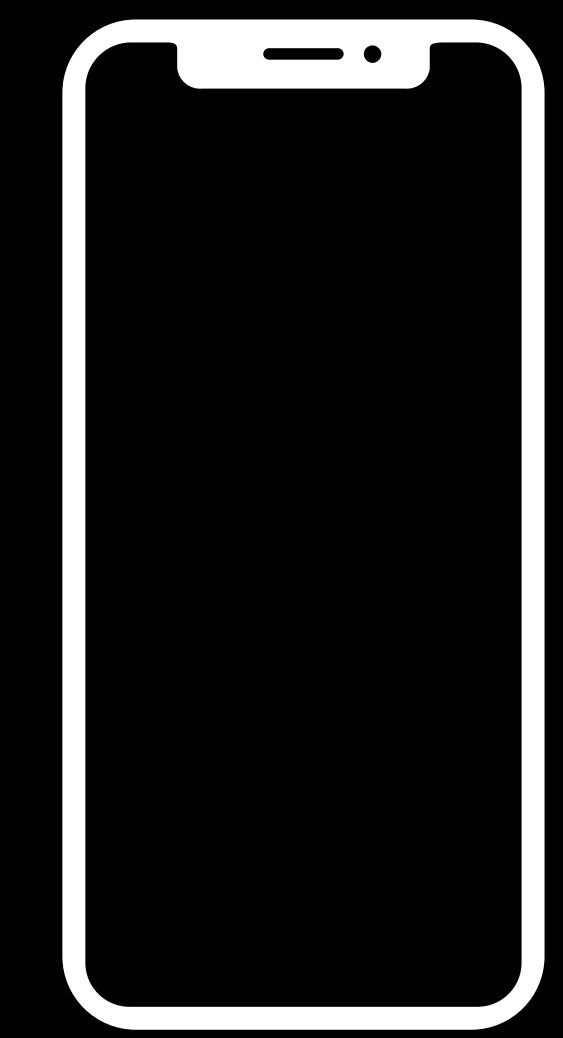


App Store

```
{ notification_type:  
  "DID_CHANGE_RENEWAL_STA  
TUS", ... }
```



Your Server



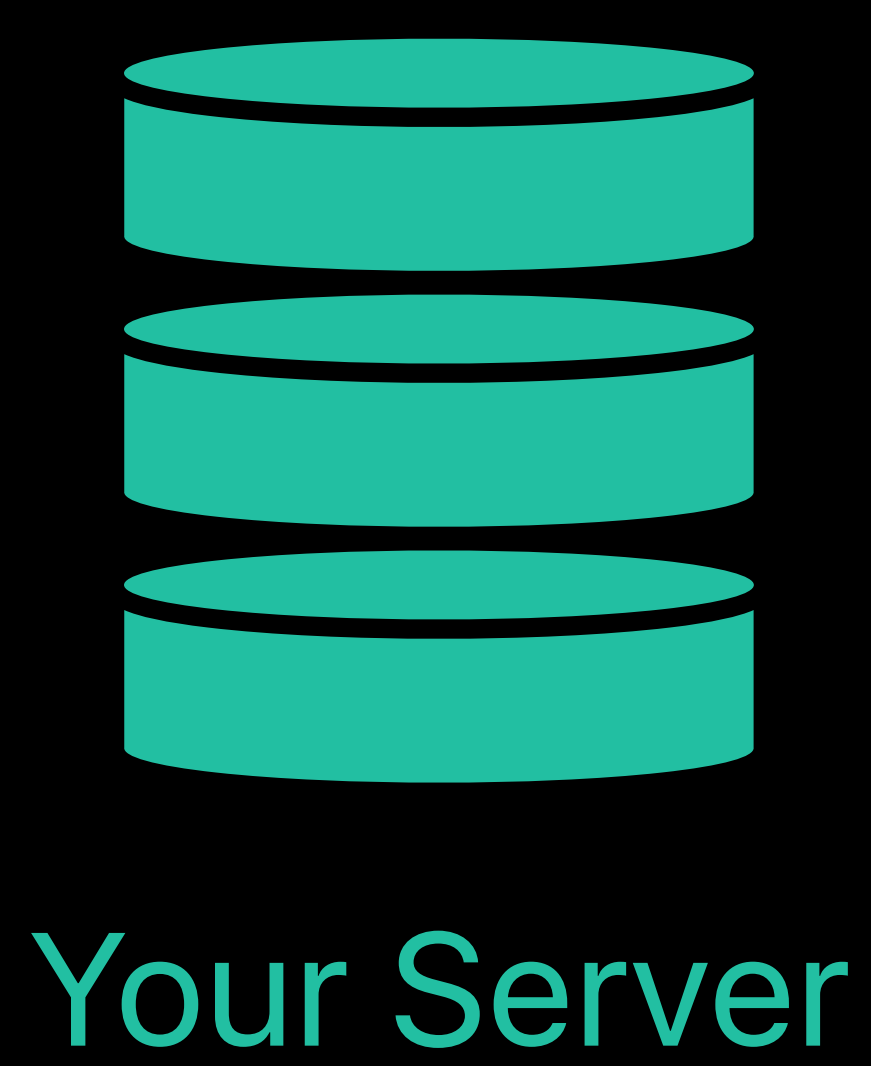
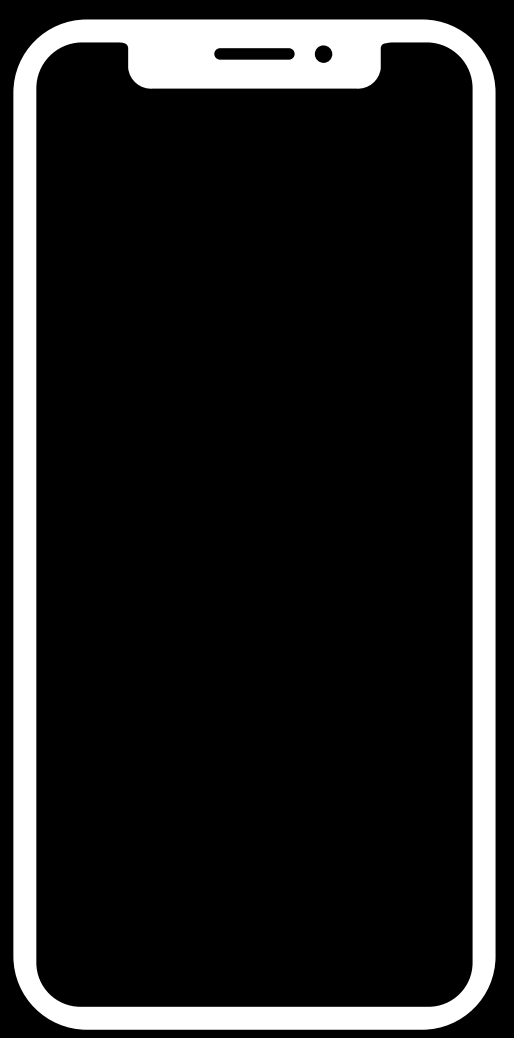
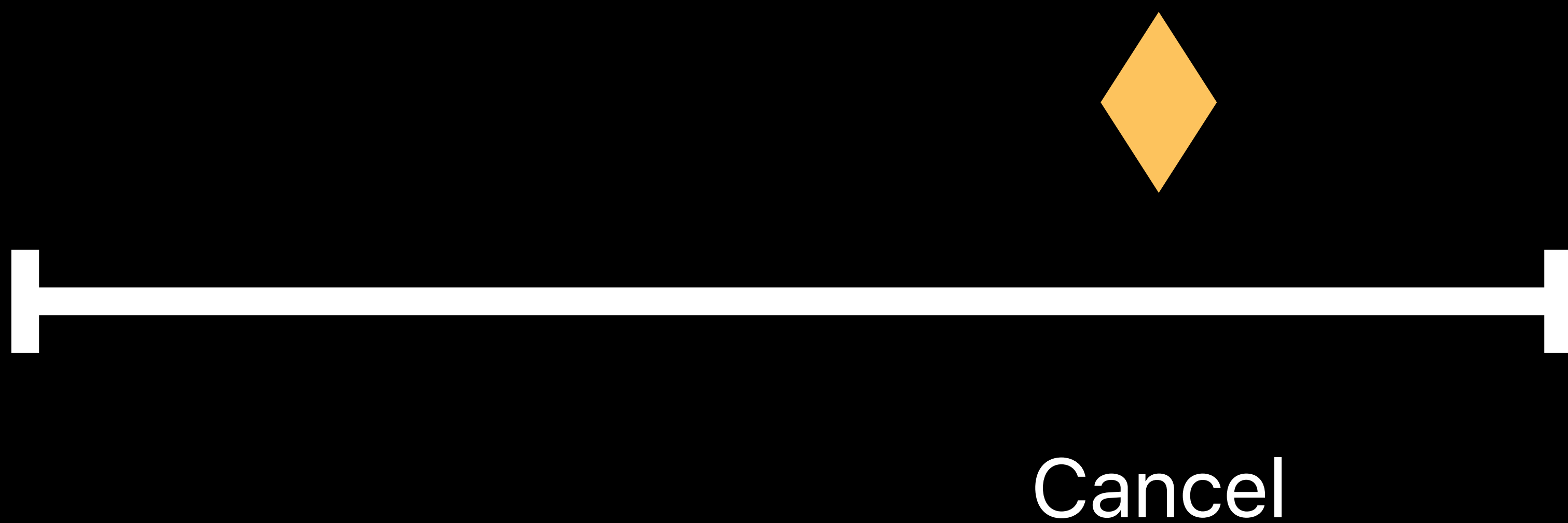
Cancel

Cancel



App Store

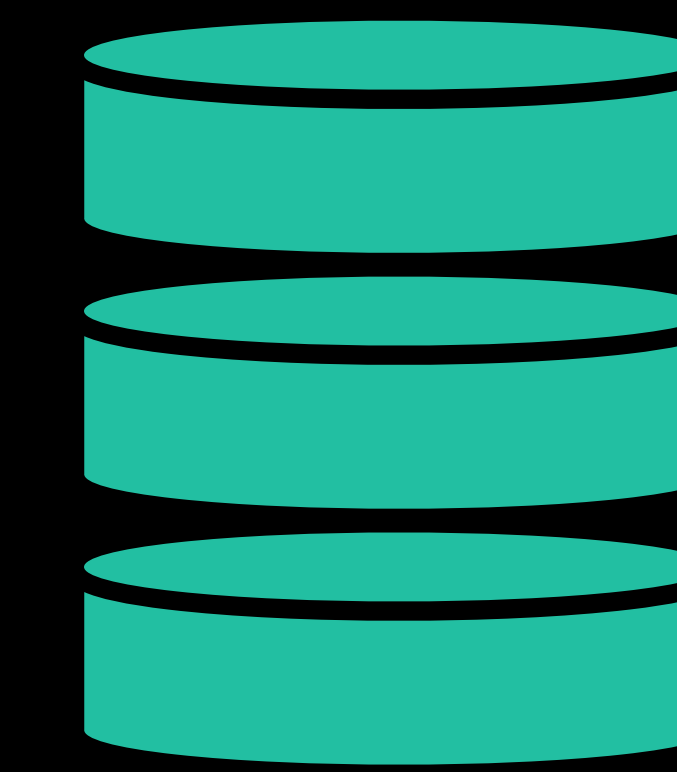
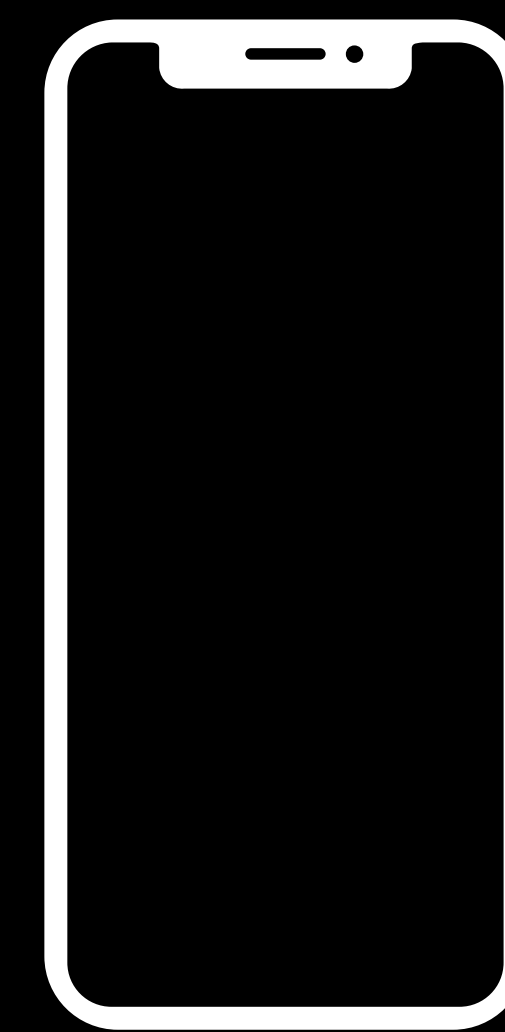
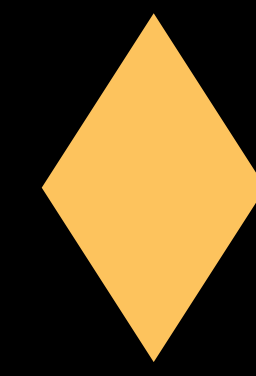
```
{ notification_type:  
  "DID_CHANGE_RENEWAL_STA  
TUS", ... }
```



userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	upgradeDate	subscriptionProductID
90000001	1133557799	2019-07-01...	TRUE	FALSE	2019-07-15...	com.product.premium

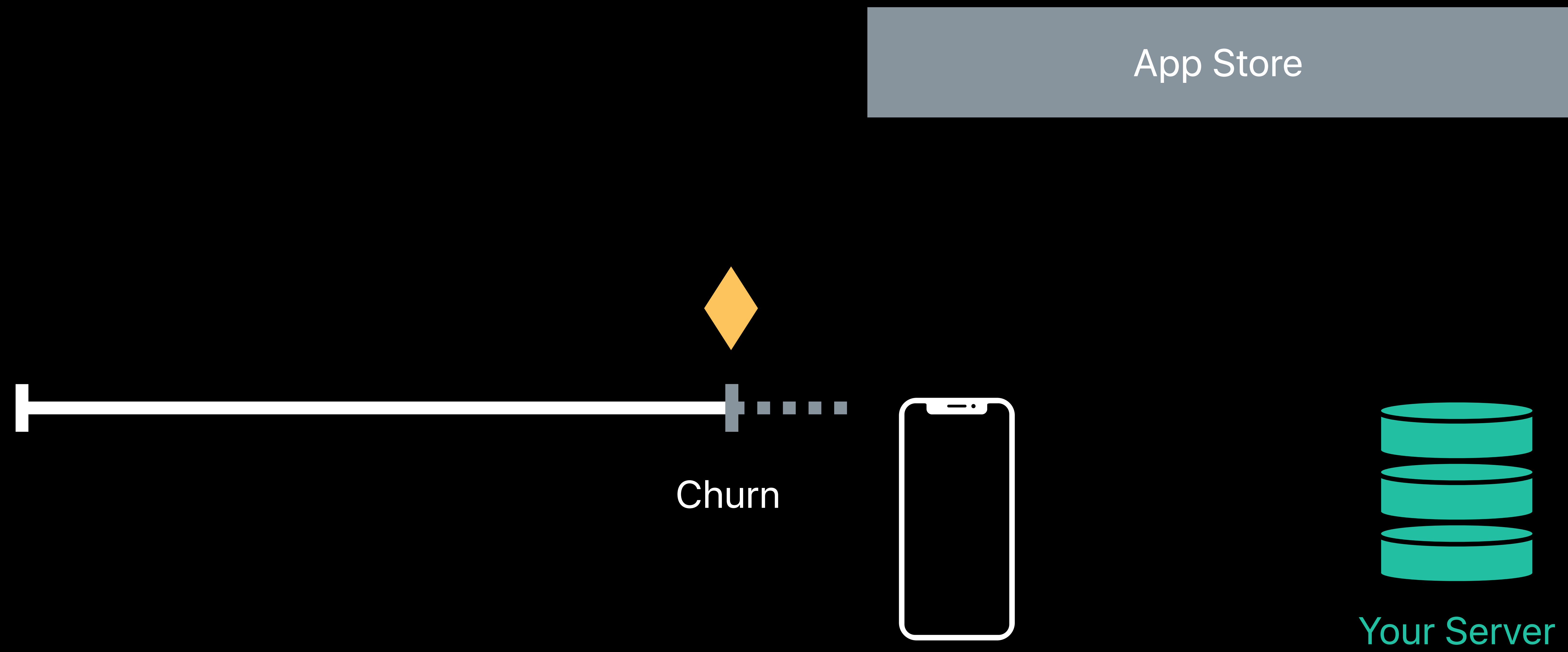
Churn

App Store

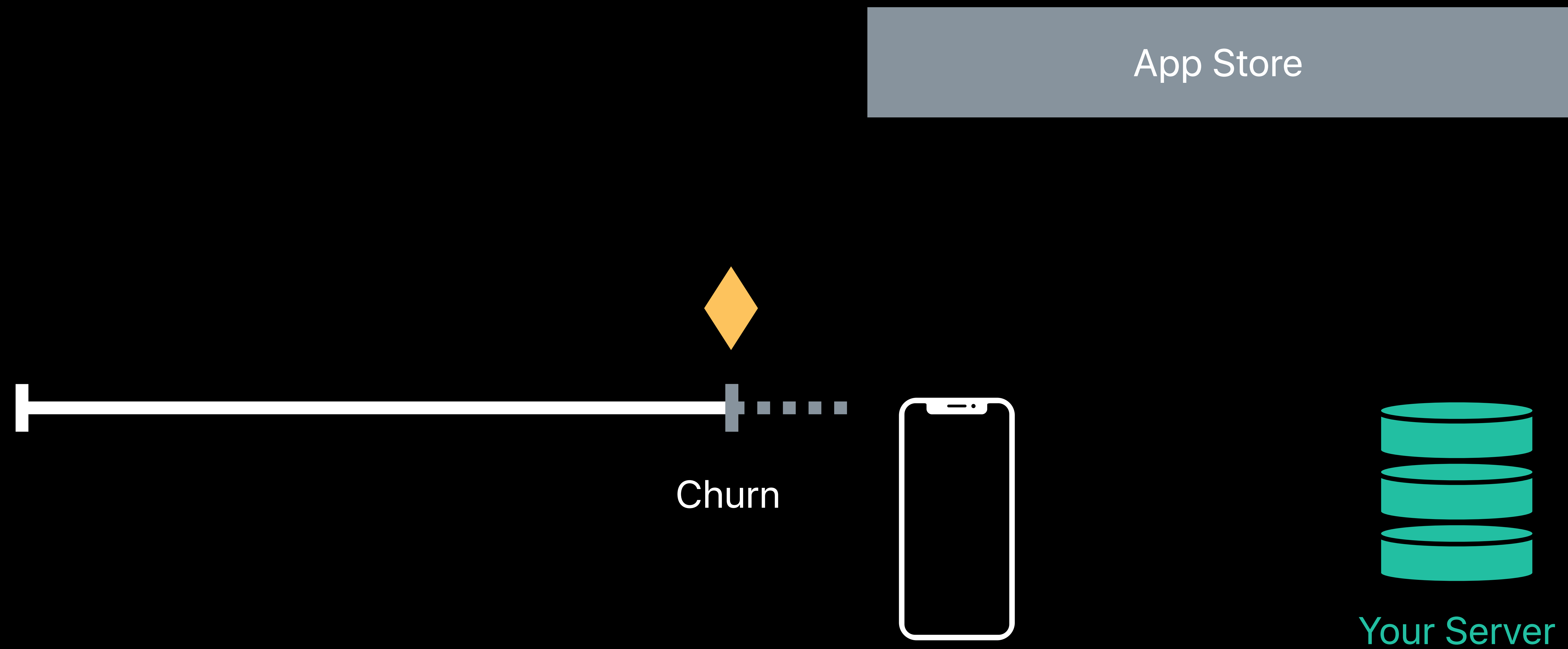


Your Server

Churn



Churn



userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	upgradeDate	subscriptionProductID
90000001	1133557799	2019-07-01...	FALSE	FALSE	2019-07-15...	com.product.premium

Win Back

Using Subscription Offers



Win Back

Using Subscription Offers



Subscription Offers Best Practices

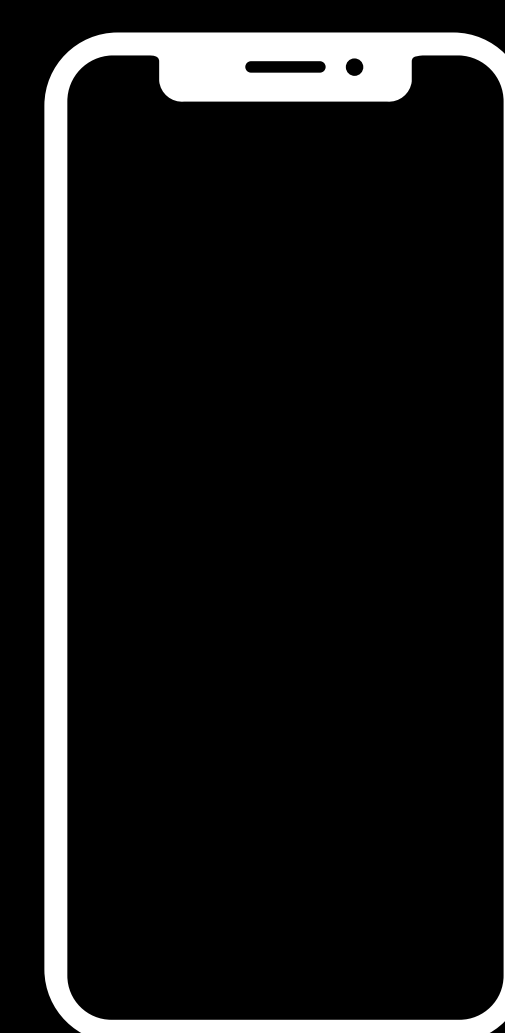
Friday, 2:00

In-App Purchases and Subscriptions Lab

Friday, 3:00

Billing Error

App Store



Your Server

Billing Error



Billing Error

NEW



Billing Error

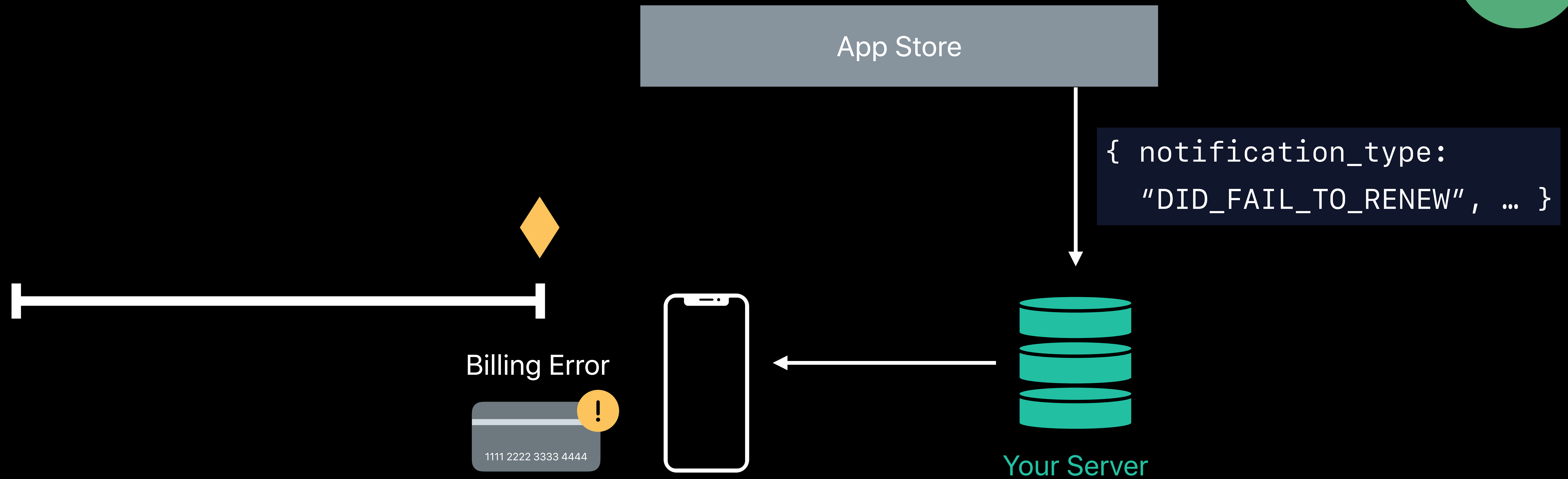
NEW



userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	subscriptionProductID
90000001	1133557799	2019-07-01...	FALSE	TRUE	com.product.premium

Billing Error

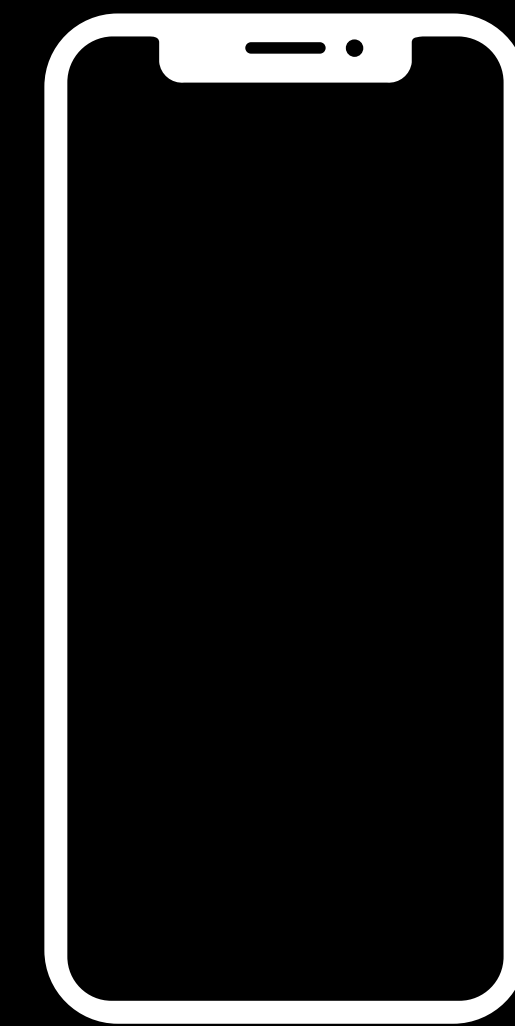
NEW



userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	subscriptionProductID
90000001	1133557799	2019-07-01...	FALSE	TRUE	com.product.premium

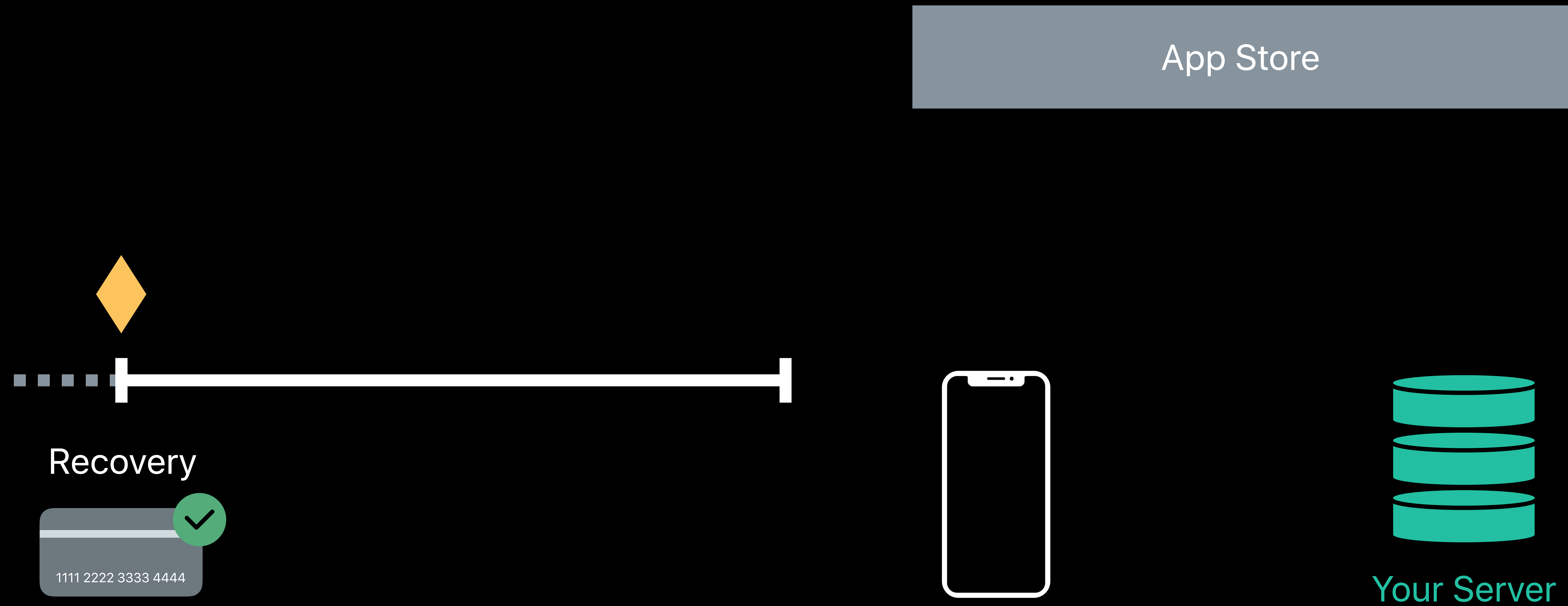
Billing Retry

App Store



Your Server

Billing Retry



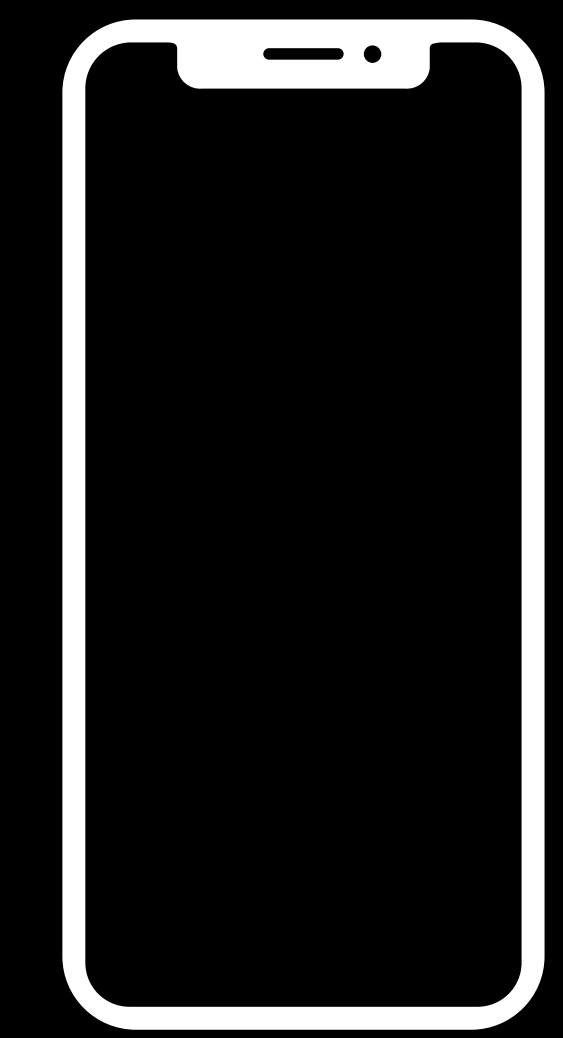
Billing Retry

NEW



App Store

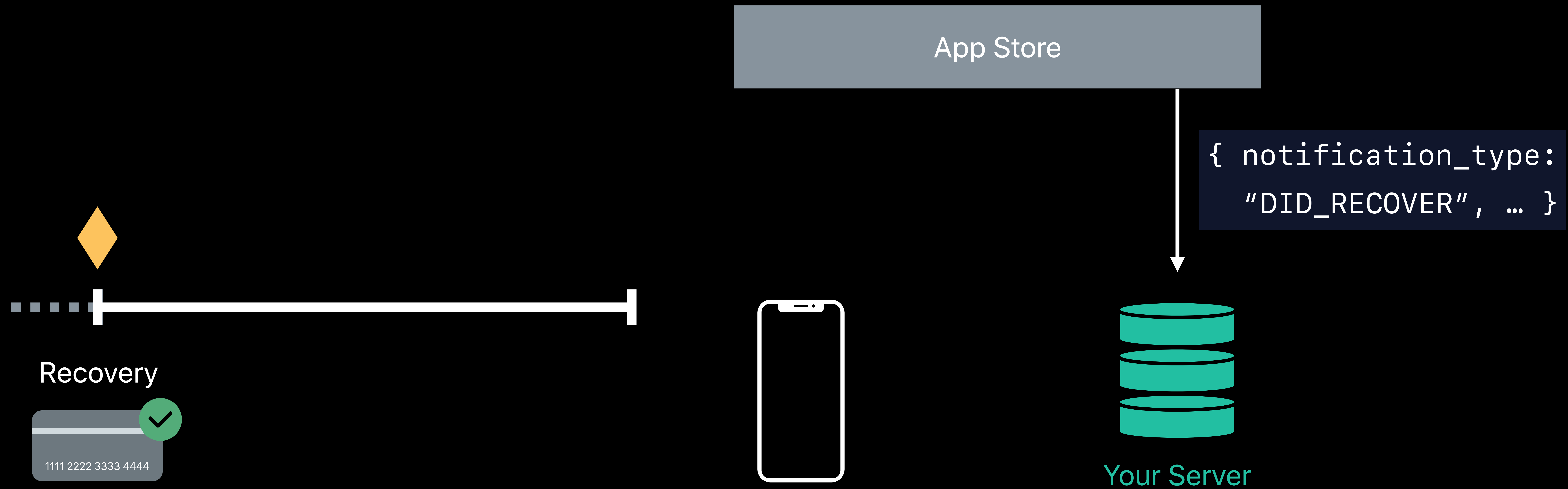
```
{ notification_type:  
  "DID_RECOVER", ... }
```



Your Server

Billing Retry

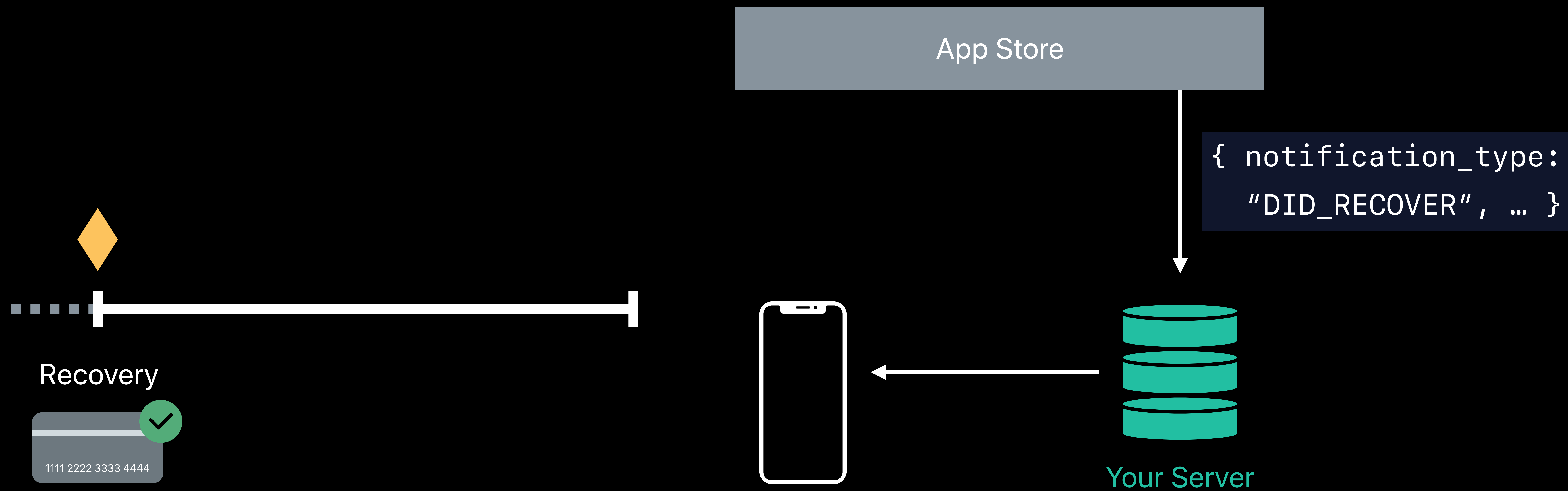
NEW



userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	subscriptionProductID
90000001	1133557799	2019-09-01...	TRUE	TRUE	com.product.premium

Billing Retry

NEW



userId	originalTransactionId	latestExpiresDate	isSubscribed	willAutoRenew	subscriptionProductID
90000001	1133557799	2019-09-01...	TRUE	TRUE	com.product.premium

Reducing Involuntary Churn

Billing Retry



Billing Retry

Extended failed payment logic



Billing Retry

Extended failed payment logic

Updated retry strategies



Billing Retry

Extended failed payment logic

Updated retry strategies

ML models

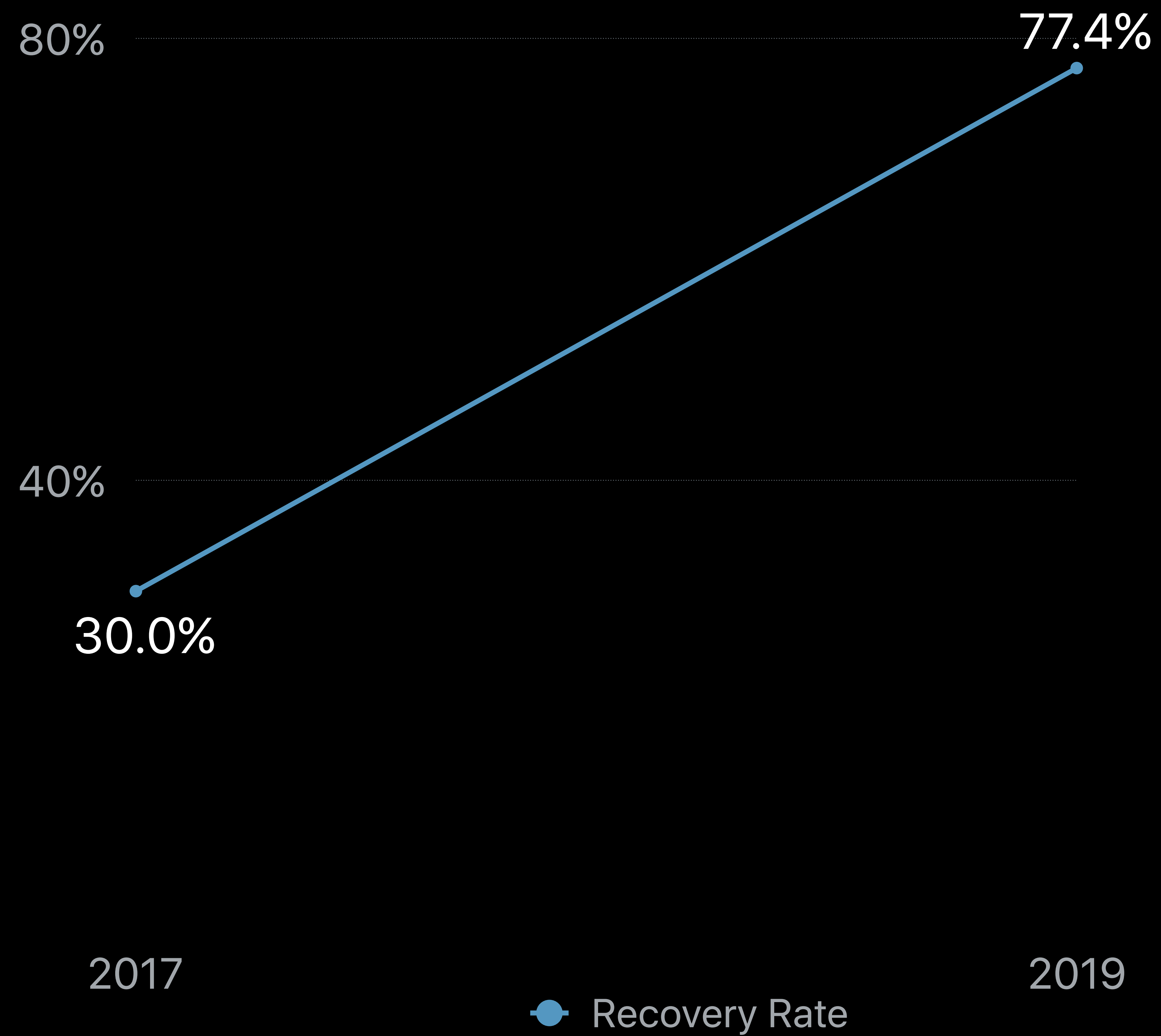


Billing Retry

Performance

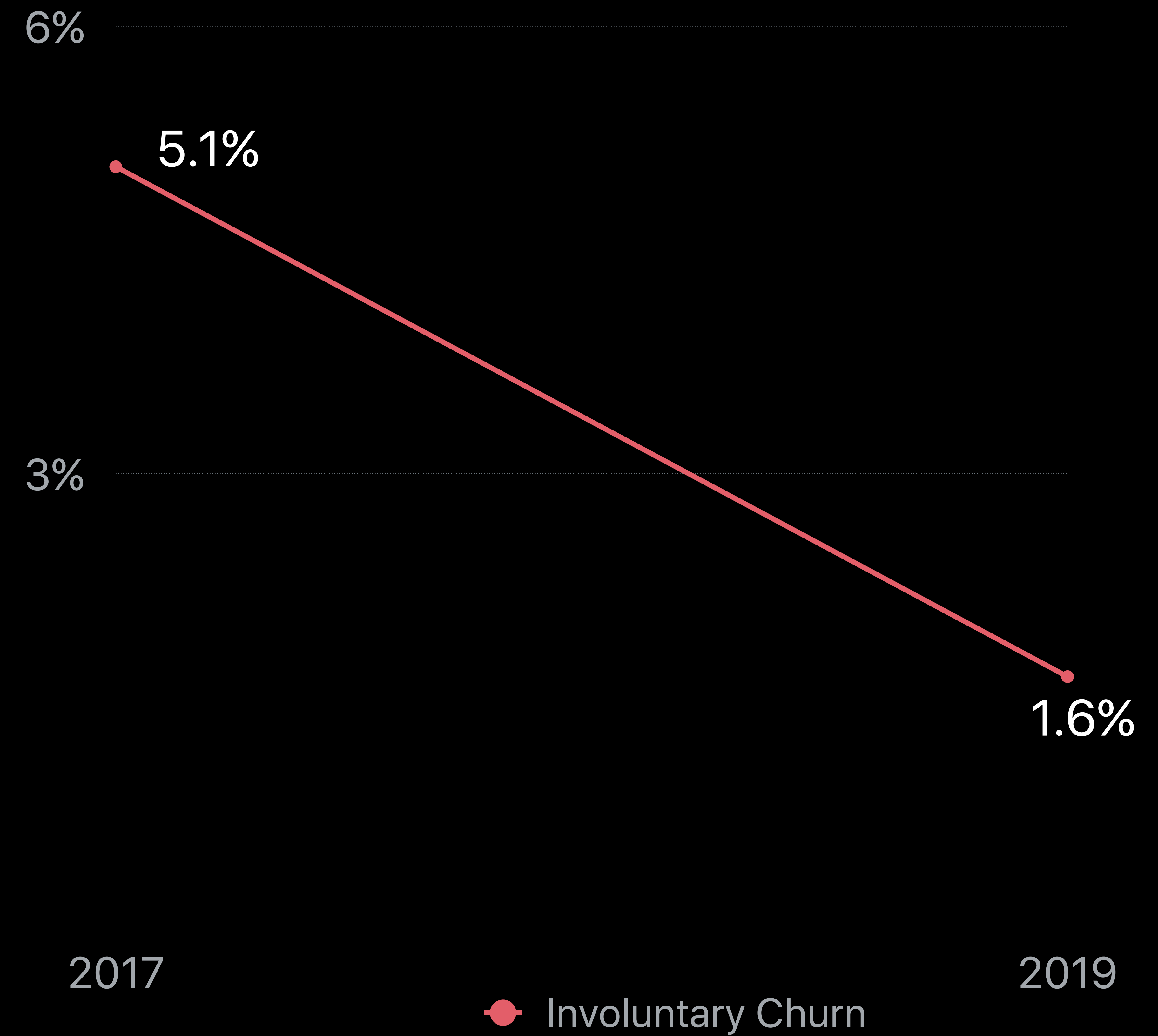
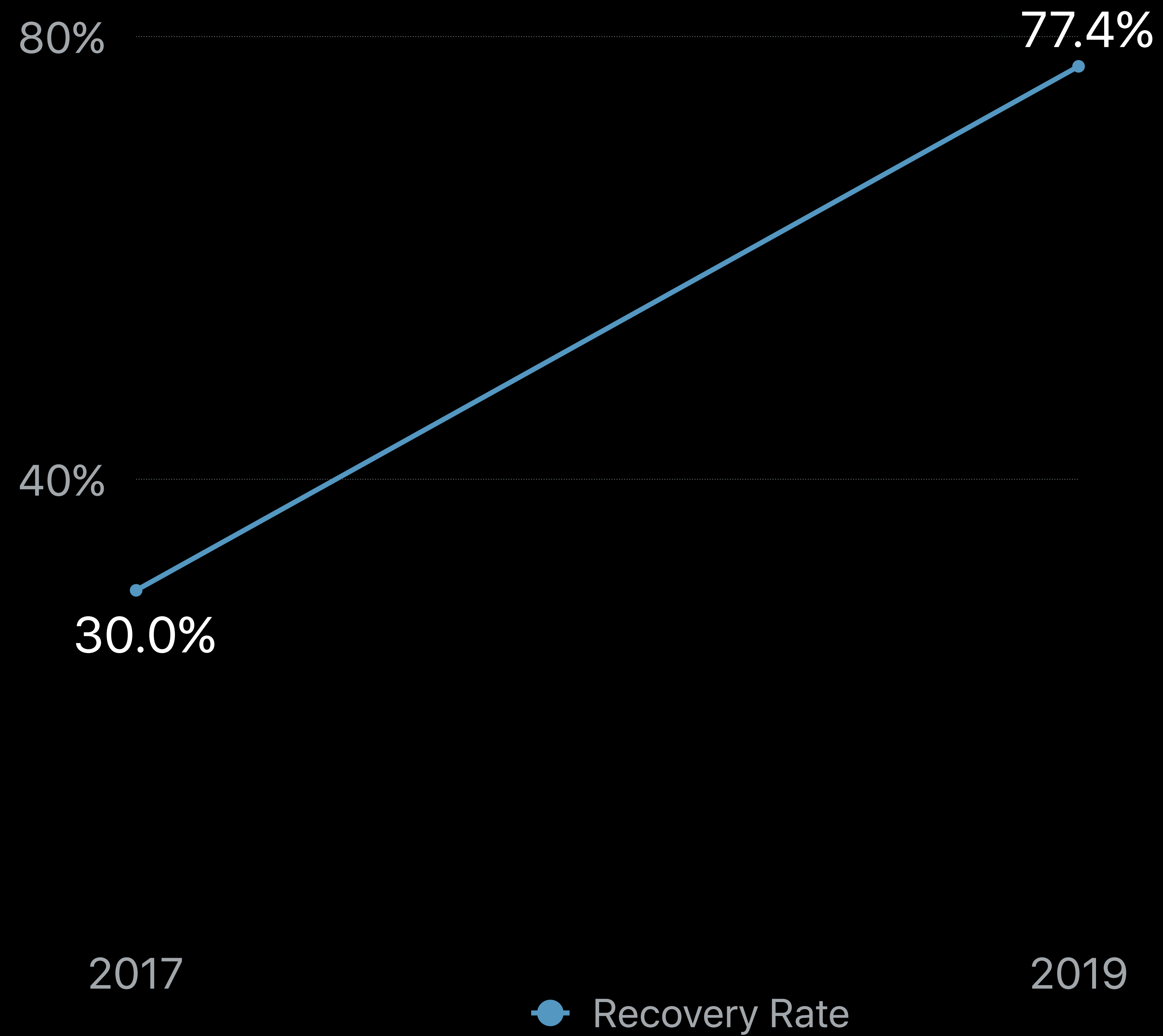
Billing Retry

Performance



Billing Retry

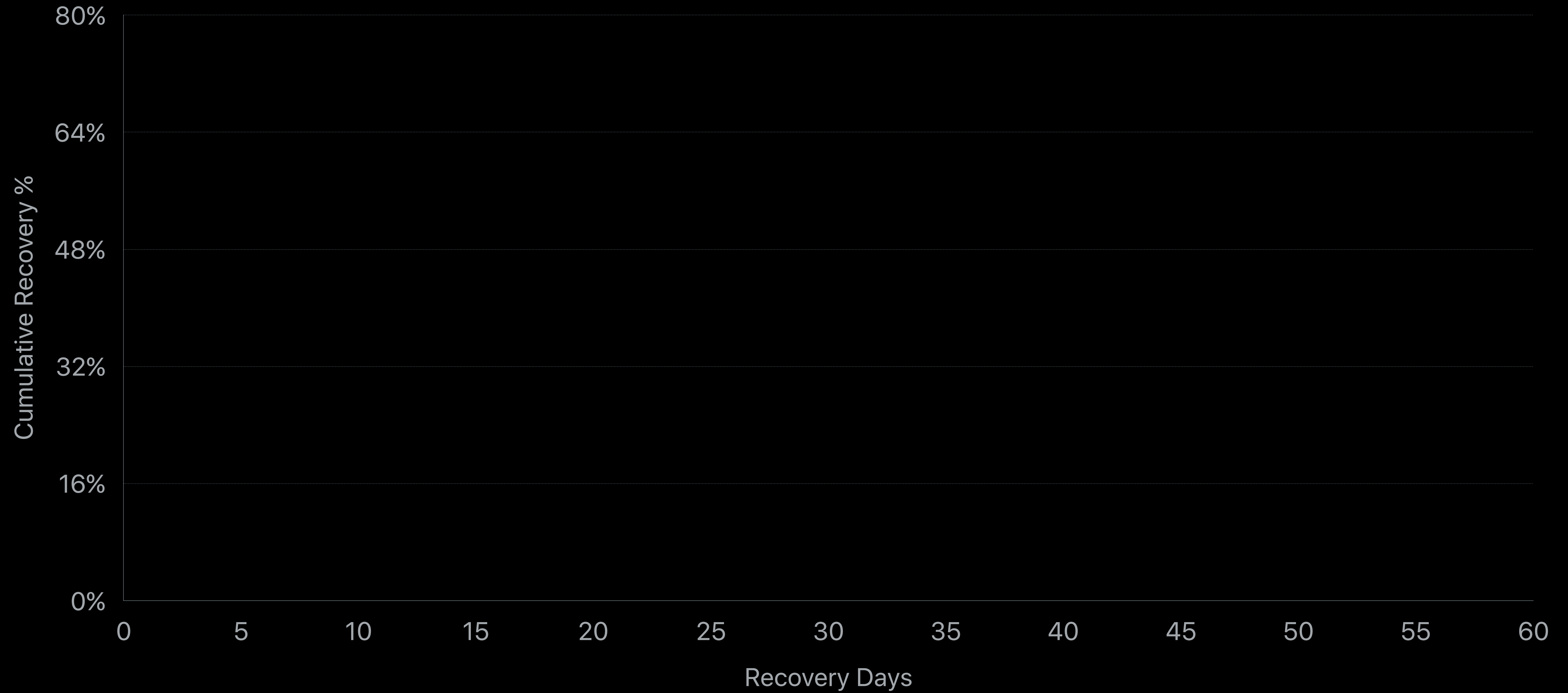
Performance



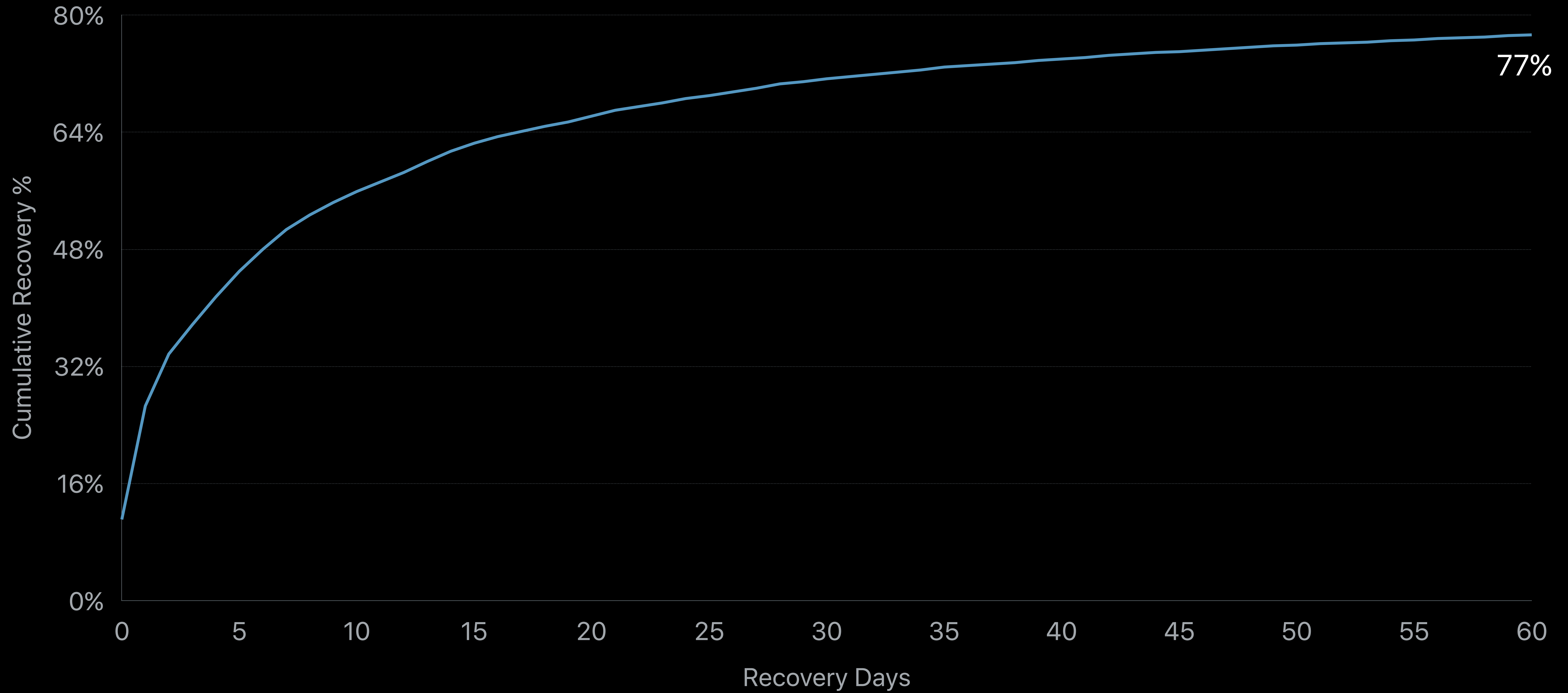
46 Million

Subscriptions recovered

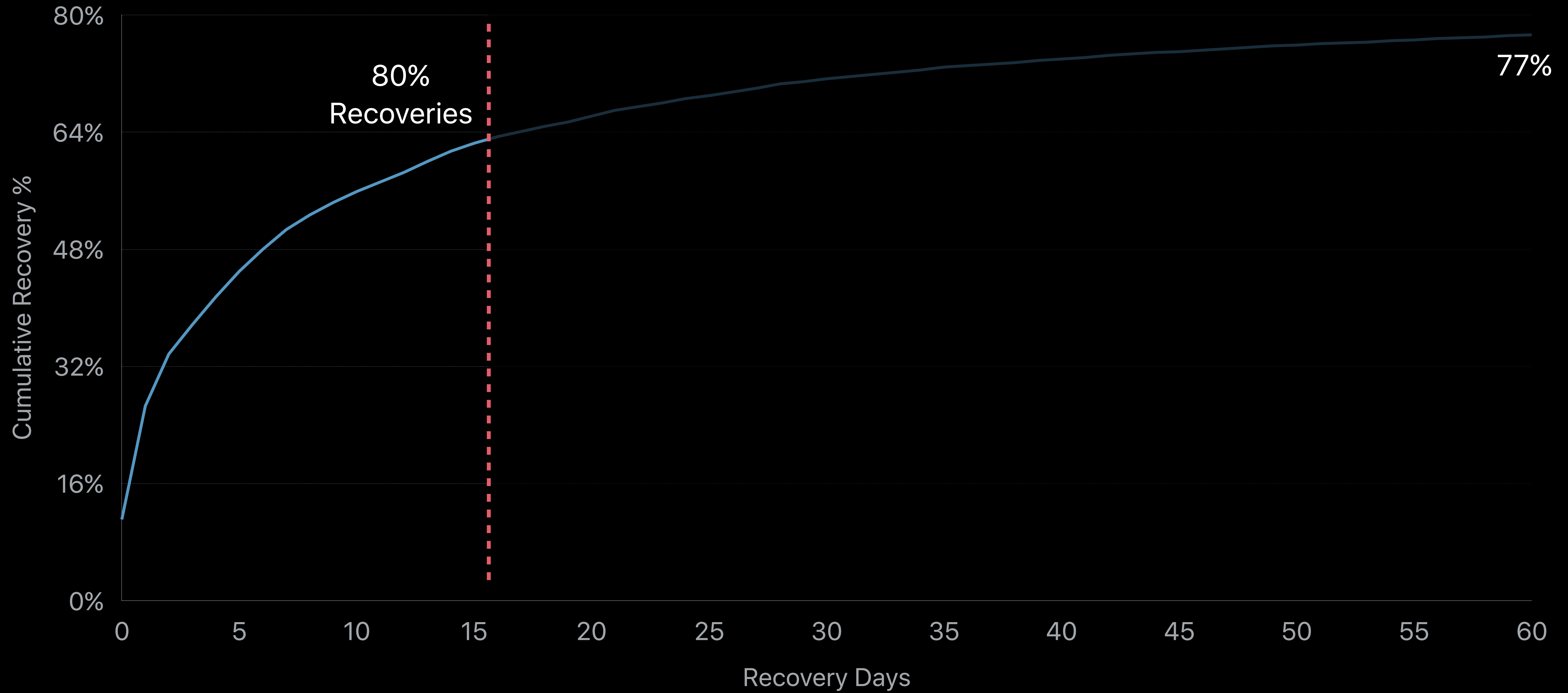
Recovery Rate



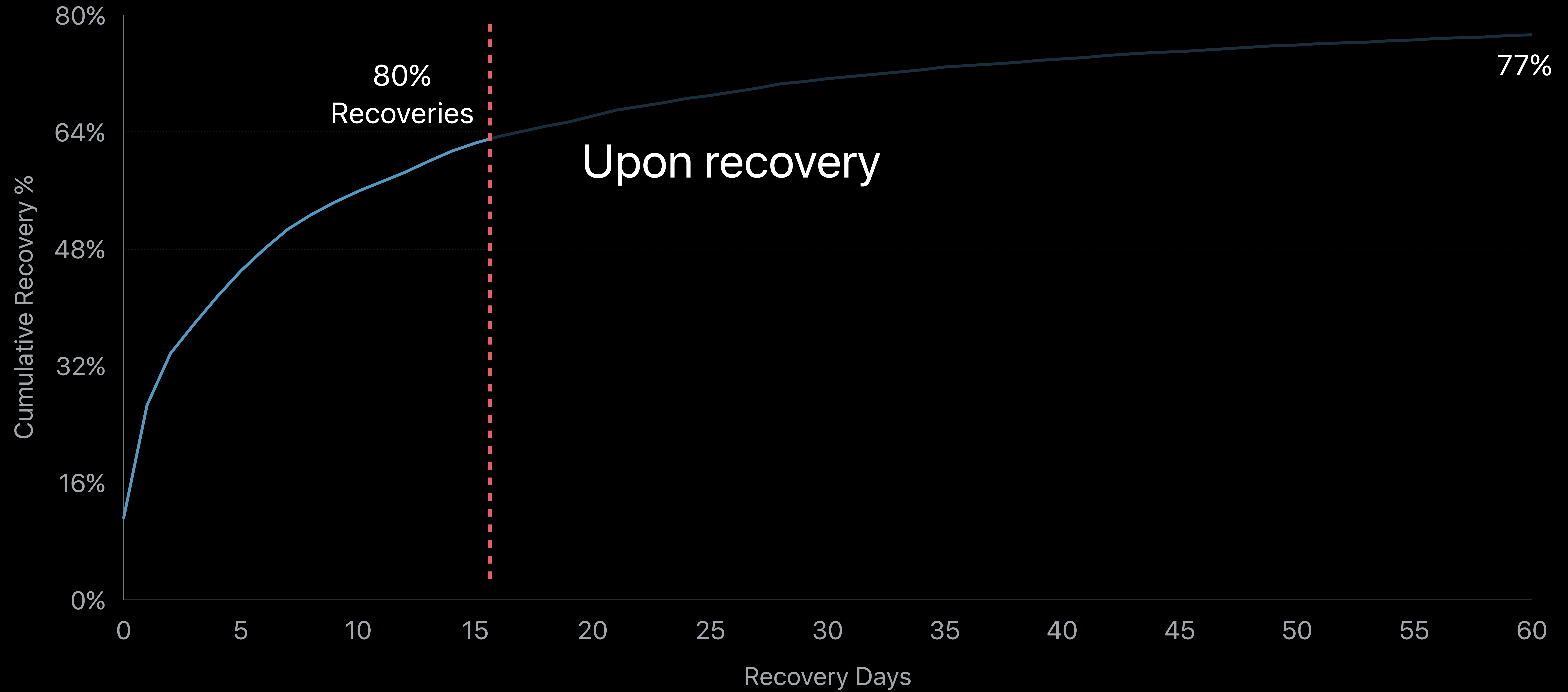
Recovery Rate



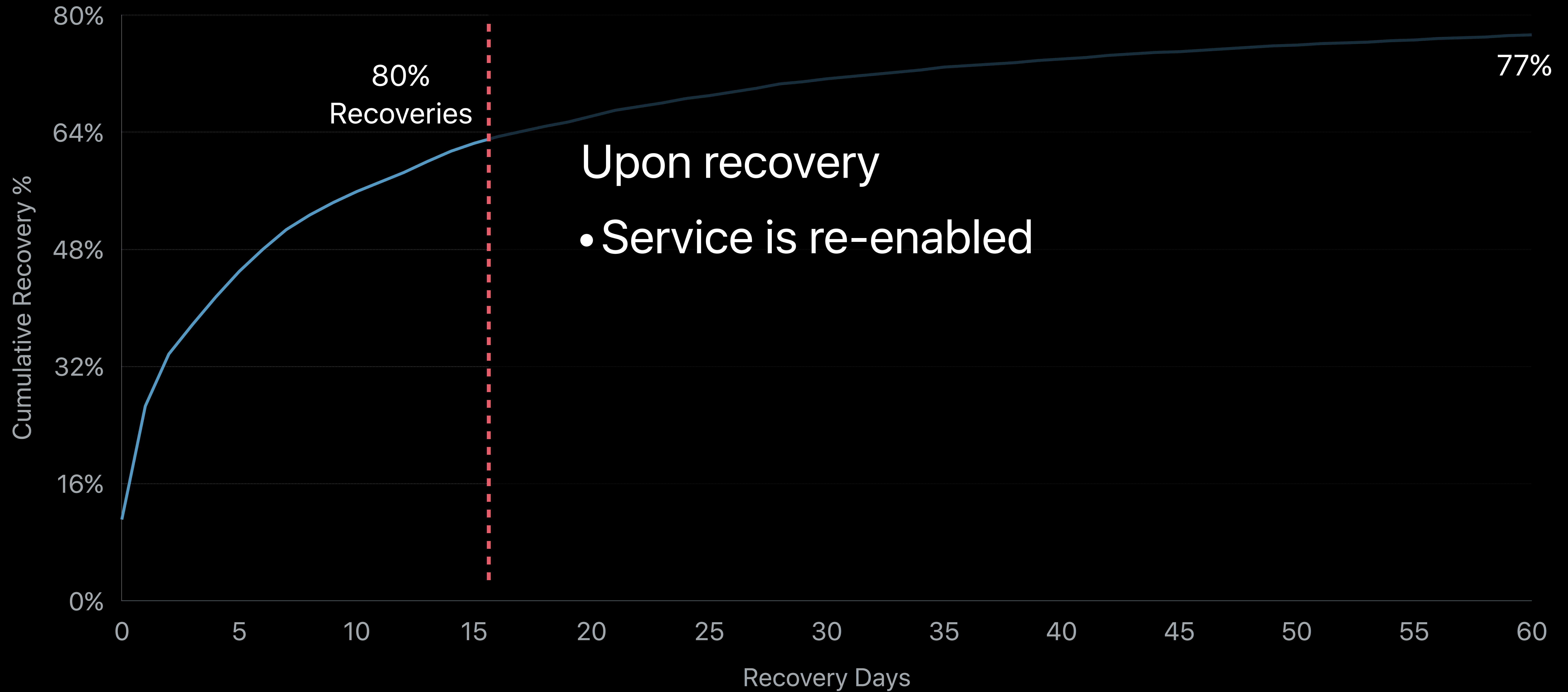
Recovery Rate



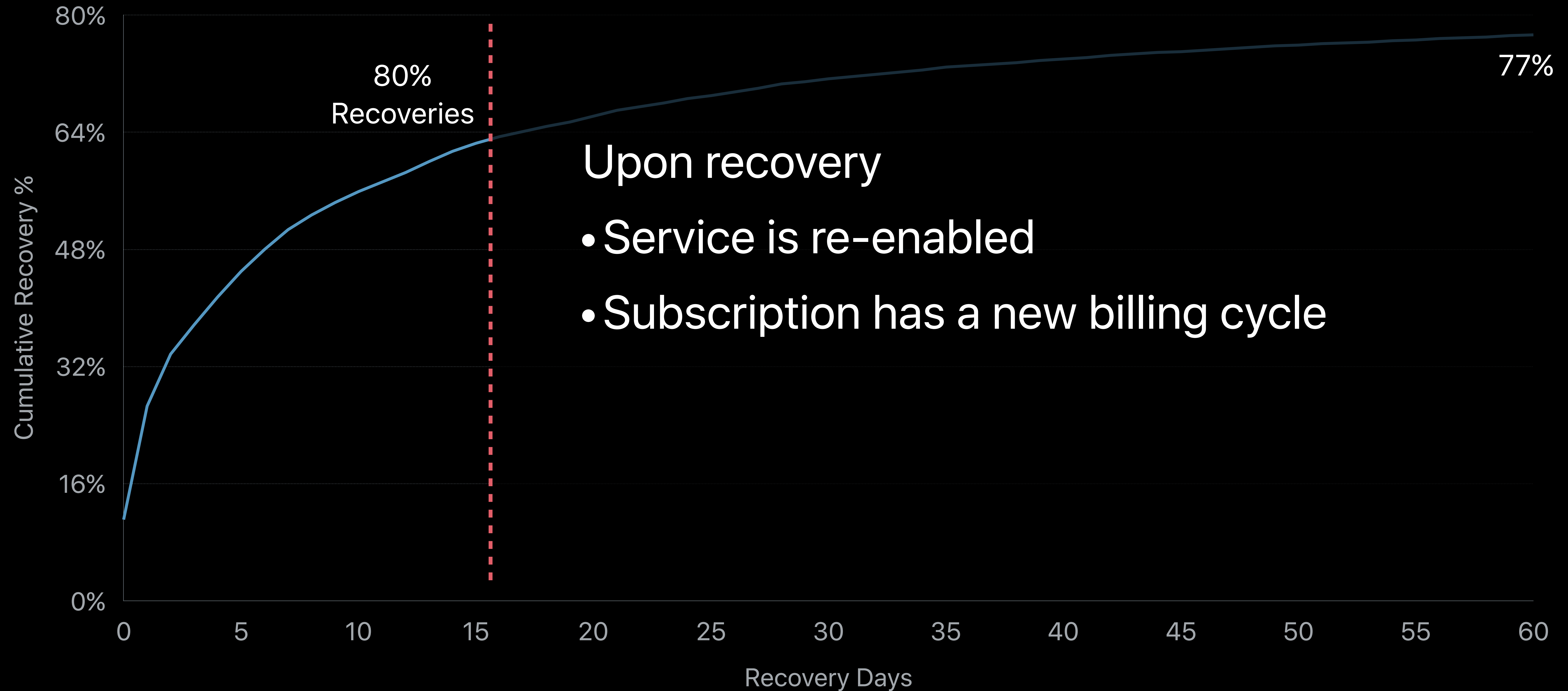
Recovery Rate



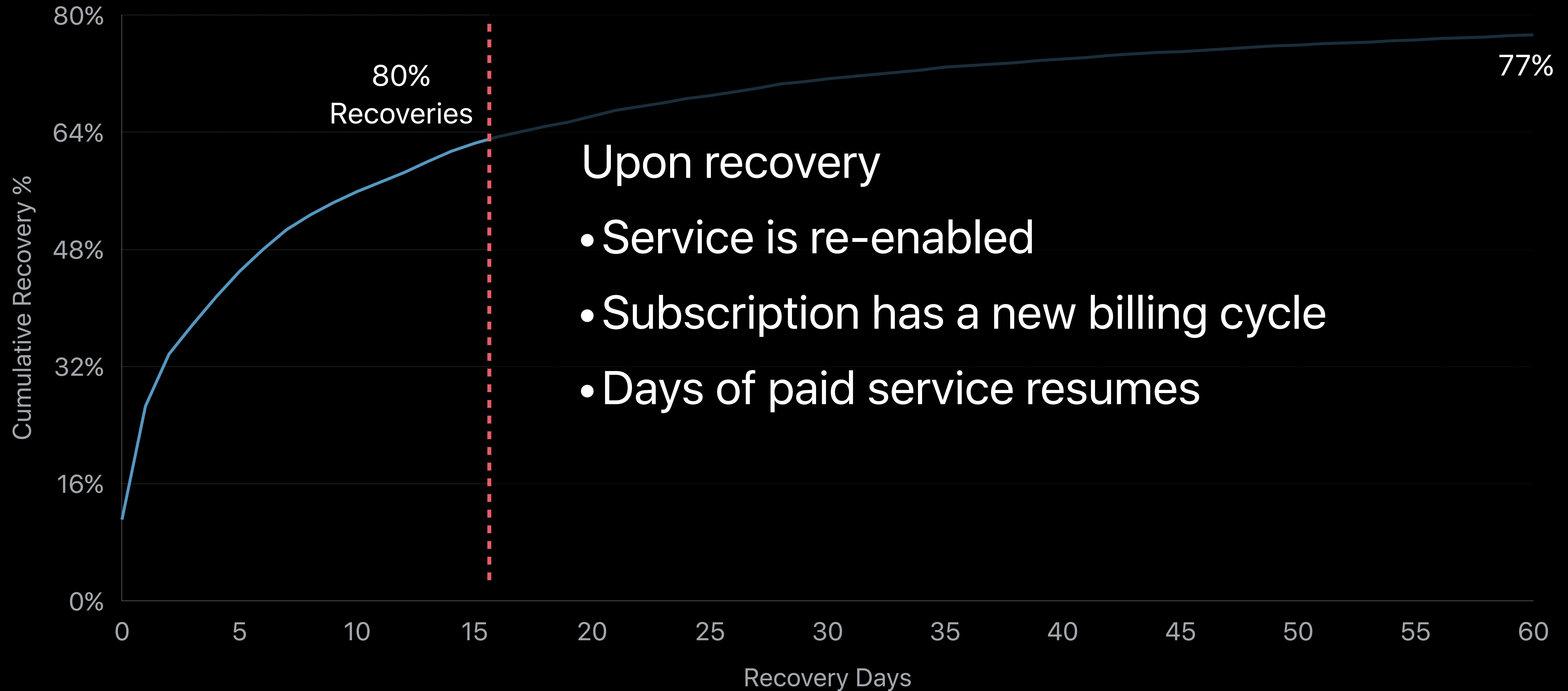
Recovery Rate



Recovery Rate



Recovery Rate



Billing Grace Period

Overview

NEW

Billing Grace Period

Overview



NEW

Starting this Fall, you will be able to offer a billing grace period, which allows

Billing Grace Period

Overview



NEW

Starting this Fall, you will be able to offer a billing grace period, which allows

- Subscribers to retain access

Billing Grace Period

Overview



NEW

Starting this Fall, you will be able to offer a billing grace period, which allows

- Subscribers to retain access
- Developers to earn additional revenue

Billing Grace Period Implementation

NEW

Billing Grace Period Implementation



NEW

1. Opt-in via App Store Connect

Billing Grace Period

Implementation



NEW

1. Opt-in via App Store Connect
2. New field in `/verifyReceipt` – `grace_period_expires_date`

Billing Grace Period

Implementation



NEW

1. Opt-in via App Store Connect
2. New field in `/verifyReceipt` – `grace_period_expires_date`
3. Keep service active during grace period

Billing Grace Period

Benefits

Billing Grace Period

Benefits

No service interruption for customers

Billing Grace Period

Benefits

No service interruption for customers

Billing cycle continuity for recoveries made within grace period

Billing Grace Period

Benefits

No service interruption for customers

Billing cycle continuity for recoveries made within grace period

Revenue for service provided during grace period

Billing Grace Period

Benefits

No service interruption for customers

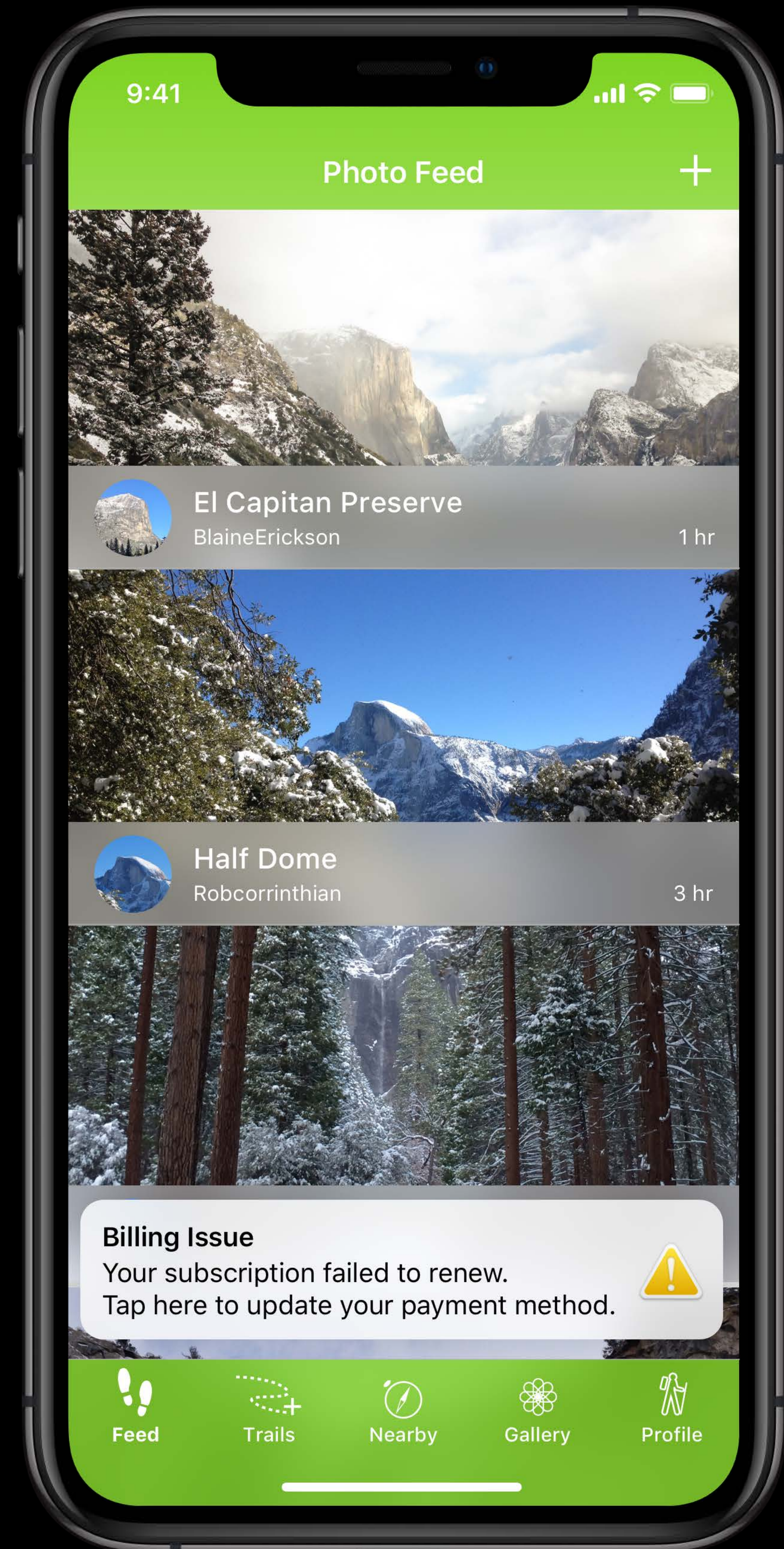
Billing cycle continuity for recoveries made within grace period

Revenue for service provided during grace period

Faster accumulation of days of paid service towards 85/15 revenue share

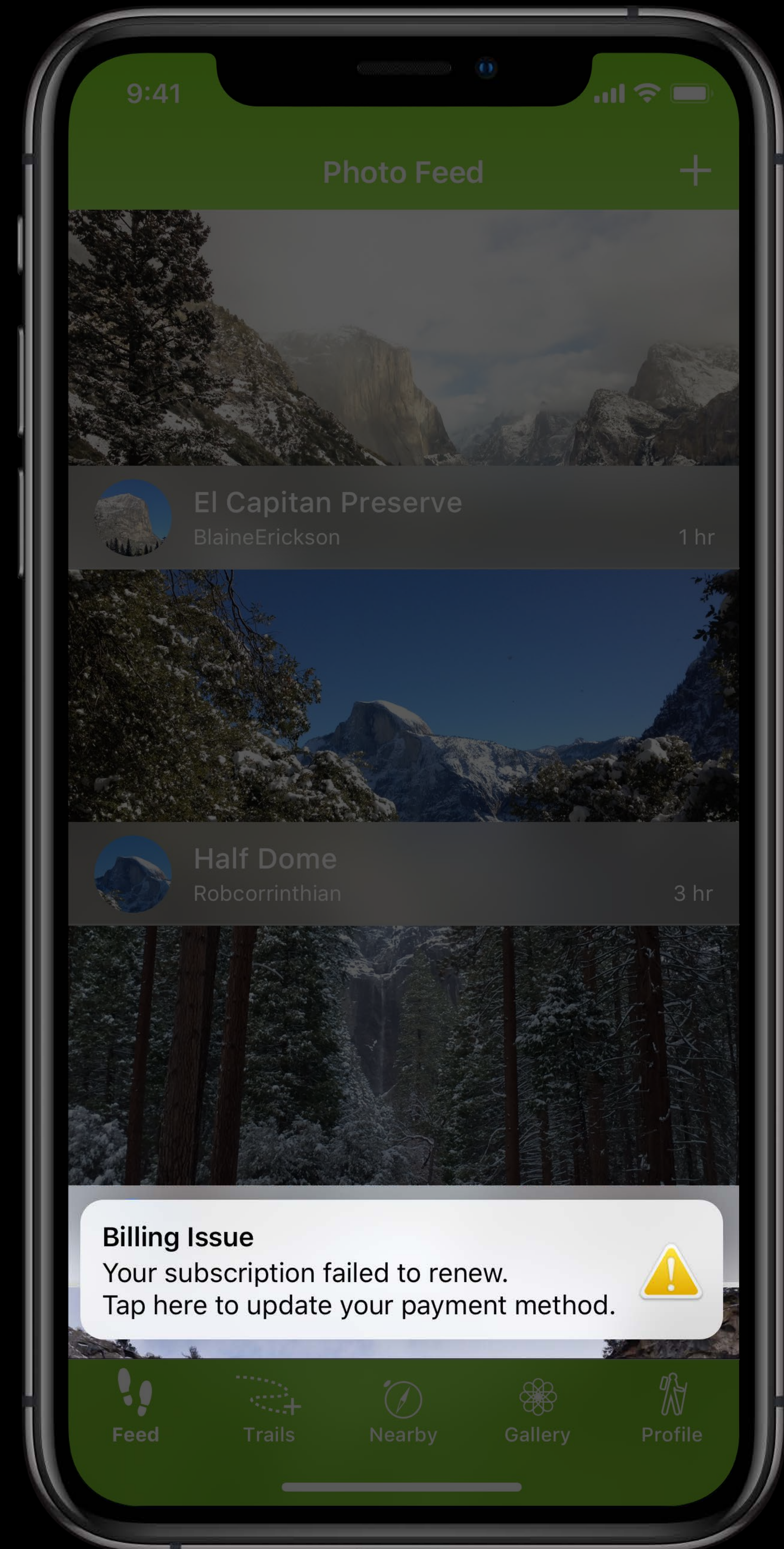
Billing Retry

Customer messaging



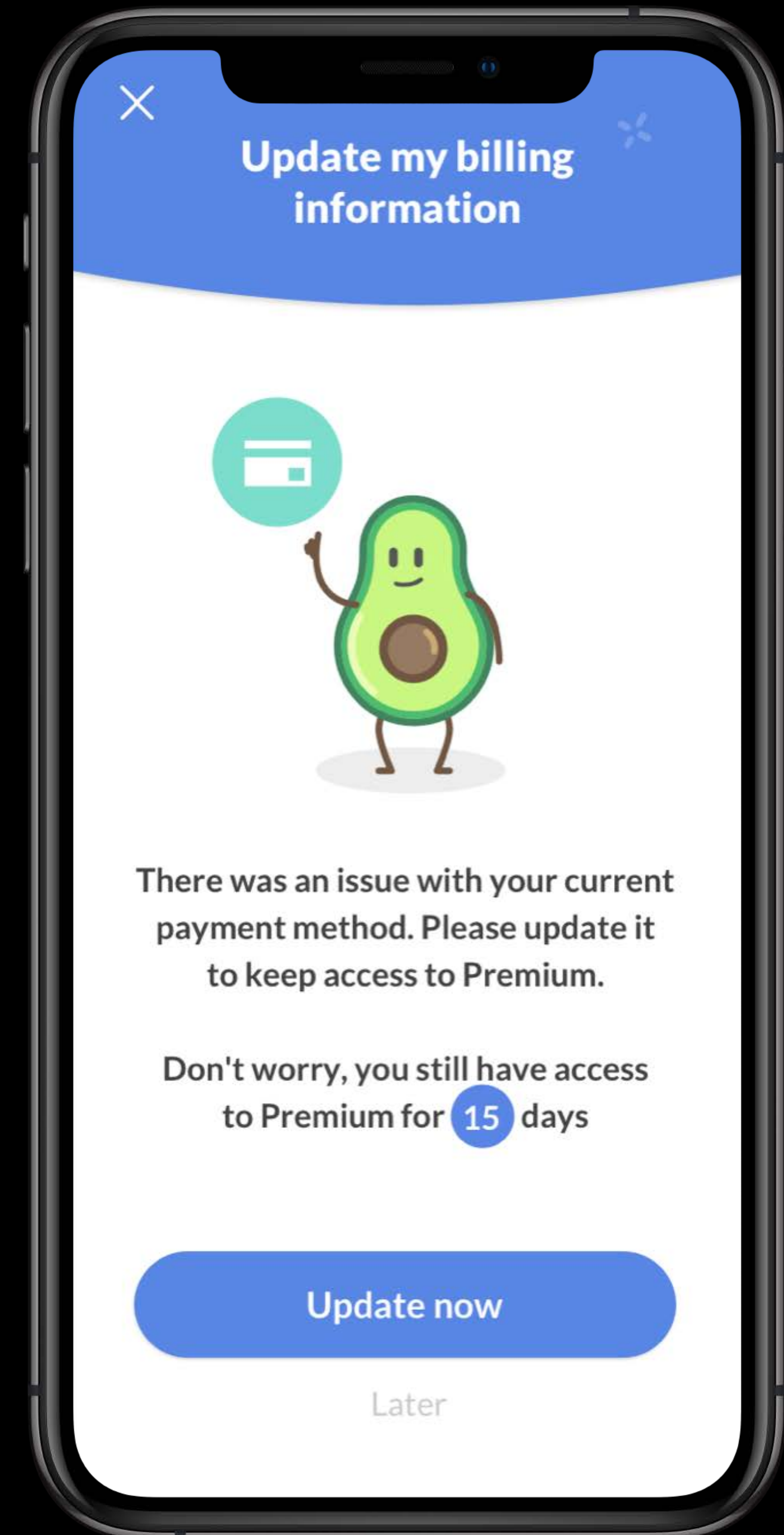
Billing Retry

Customer messaging



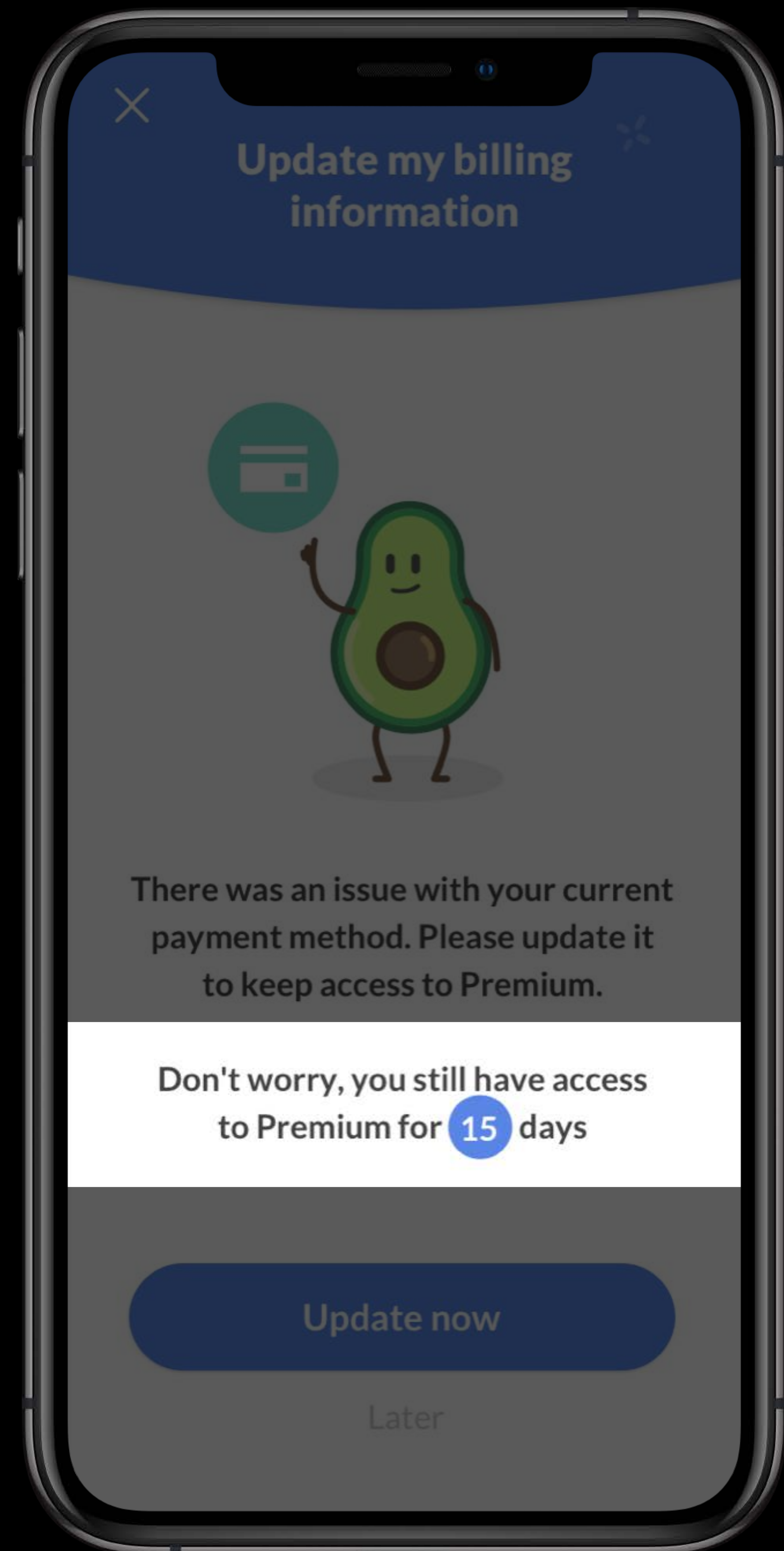
Billing Retry

Customer messaging



Billing Retry

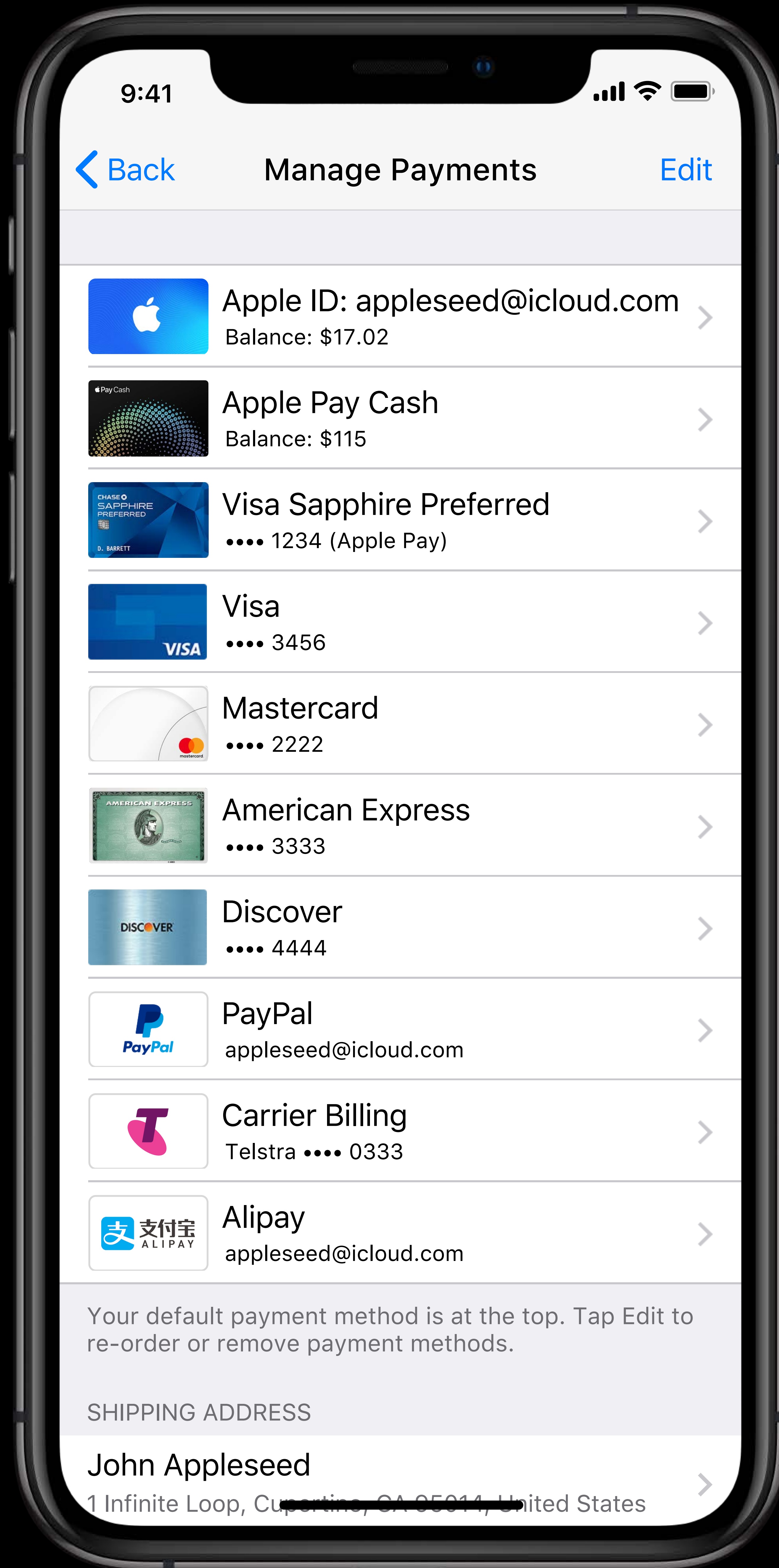
Customer messaging



Billing Grace Period

Customer messaging

Edit billing information — apps.apple.com/account/billing



NEW

Summary

Summary

Reduce voluntary churn by using Subscription Offers

Summary

Reduce voluntary churn by using Subscription Offers

Implement SKStorefront to show the right content to the customer

Summary

Reduce voluntary churn by using Subscription Offers

Implement SKStorefront to show the right content to the customer

Identify customers who pre-ordered your app

Summary

Reduce voluntary churn by using Subscription Offers

Implement SKStorefront to show the right content to the customer

Identify customers who pre-ordered your app

Use server-to-server notifications and respond to billing events

Summary

Reduce voluntary churn by using Subscription Offers

Implement SKStorefront to show the right content to the customer

Identify customers who pre-ordered your app

Use server-to-server notifications and respond to billing events

Provide a billing grace period

Summary

Reduce voluntary churn by using Subscription Offers

Implement SKStorefront to show the right content to the customer

Identify customers who pre-ordered your app

Use server-to-server notifications and respond to billing events

Provide a billing grace period

Show contextual messaging inside the app

More Information

developer.apple.com/wwdc19/302

Subscription Offers Best Practices

Friday, 2:00

In-App Purchases and Subscriptions Lab

Friday, 3:00

