Agenda
Agenda

Overview

Assigning Tags

Using BundleResourceRequest

Optimizing First Launch

Optimizing Predictive Loading

Optimizing App Updates
Motivation
Traditional App

- App Binary
- Resources
Traditional App

App Bundle

App Binary

Resources
Traditional App
Traditional App
Traditional App
On-Demand Resources App

App Bundle

Bundled Resources

App Binary
On-Demand Resources App

App Bundle

- App Binary
- Bundled Resources

On-Demand Resources
On-Demand Resources App

- **App Binary**: Up to 200MB (tvOS) / Up to 4GB (iOS)
- **Bundled Resources**: Up to 20GB (tvOS) / Up to 20GB (iOS)

**App Bundle**
On-Demand Resources

Dynamically loaded content
On-Demand Resources

Dynamically loaded content
Hosted on the App Store
On-Demand Resources

Dynamically loaded content
Hosted on the App Store
Downloadable during app install and by request
On-Demand Resources

Dynamically loaded content
Hosted on the App Store
Downloadable during app install and by request
Prioritized downloads
On-Demand Resources

- Dynamically loaded content
- Hosted on the App Store
- Downloadable during app install and by request
- Prioritized downloads
- Intelligent content caching
On-Demand Resources

Benefits

Smaller main app bundle
• Faster initial download
On-Demand Resources

Benefits

Smaller main app bundle
• Faster initial download

Richer app content
• Up to 20-GB available on demand
On-Demand Resources

Benefits

Smaller main app bundle
- Faster initial download

Richer app content
- Up to 20-GB available on demand

More apps installed and ready to run
- Reduces need to manage storage
Assigning Tags
Assigning Tags
Packaging for download

Organize your resources
• By role within app
• By when you need them
Assigning Tags
Packaging for download

Organize your resources
- By role within app
- By when you need them

Apply tags using Xcode
- Like "Level 1"
- Tags are simple strings
- May apply to single assets or entire folders
Resources

GreatGame.app
Resources
GreatGame.app
Resources
GreatGame.app

App Binary

- Required Always
- Level One
- Level Two
- Level Three
- Purchasable Item
Resources

GreatGame.app

App Binary

Required Always
Level One
Level Two
Level Three
Purchasable Item

Include in App Bundle
“Level 01”
“Level 02”
“Level 03”
“Item 01”
Resources

GreatGame.app

App Binary

- Required Always
  - Level One
  - Level Two
  - Level Three
  - Purchasable Item

Include in App Bundle

- "Level 01"
- "Level 02"
- "Level 03"
- "Item 01"
Assigning Tags

Getting started

Only bundle what’s always required

• Assets required throughout app
• UI elements required every launch
Assigning Tags

Getting started

Only bundle what’s always required
- Assets required throughout app
- UI elements required every launch

Apply tags to the rest
- Up to 512MB per tag (64MB recommended)
- More than one tag per resource, if needed
Using Tagged Resources
BundleResourceRequest

Overview

Manages access to on-demand resources

• Set up with tags and other options
• Begin and end accessing resources
• Set priority, track progress, handle errors
BundleResourceRequest

Overview

Manages access to on-demand resources

• Set up with tags and other options
• Begin and end accessing resources
• Set priority, track progress, handle errors

Create as many as you need

• Each request is one-shot
BundleResourceRequest

Overview

Manages access to on-demand resources

• Set up with tags and other options
• Begin and end accessing resources
• Set priority, track progress, handle errors

Create as many as you need

• Each request is one-shot

Request decoupled from use of resources
BundleResourceRequest

Core methods
BundleResourceRequest

Core methods

Initialize with set of tags

```swift
let request = BundleResourceRequest(tags: ["Level1"])
```
BundleResourceRequest
Core methods

Initialize with set of tags

```swift
let request = BundleResourceRequest(tags: ["Level1"])
```

Begin a request

```swift
request.beginAccessingResources { (error: NSError?) in
    ...
}
```
BundleResourceRequest

Core methods

Initialize with set of tags

```swift
let request = BundleResourceRequest(tags: ["Level1"])
```

Begin a request

```swift
request.beginAccessingResources { (error: NSError?) in
...
}
```

Access resources

```swift
var mapPath = request.bundle.pathForResource("Level1", ofType: "map")
```
BundleResourceRequest
Core methods

Initialize with set of tags

let request = BundleResourceRequest(tags: ["Level1"])

Begin a request

request.beginAccessingResources { (error: NSError?) in
    ...
}

Access resources

var mapPath = request.bundle.pathForResource("Level1", ofType: "map")

Tell the system you're finished

request.endAccessingResources()
BundleResourceRequest

Loading priority
BundleResourceRequest

Loading priority

Set relative priority of simultaneous requests

```swift
var loadingPriority: Double
```

- Value ranges from 0.0 to 1.0
BundleResourceRequest
Loading priority

Set relative priority of simultaneous requests

```swift
var loadingPriority: Double
```

- Value ranges from 0.0 to 1.0

Special urgent priority

```swift
request.loadingPriority = NSBundleResourceRequestLoadingPriorityUrgent
```

- Suspends other downloads
- Maximized throughput (and CPU consumption)
BundleResourceRequest

Conditional requests
BundleResourceRequest

Conditional requests

To check if content is present

```swift
request.conditionallyBeginAccessingResources { (available: Bool) in

  ...

}
```
BundleResourceRequest
Conditional requests

To check if content is present

```swift
request.conditionallyBeginAccessingResources { (available: Bool) in
...
}
```

If already downloaded, then identical to `beginAccessingResources()`
BundleResourceRequest
Conditional requests

To check if content is present

```swift
request.conditionallyBeginAccessingResources { (available: Bool) in
...
}
```

If already downloaded, then identical to `beginAccessingResources()`

Always call `endAccessingResources()`
Optimizing First Launch
Launch Timeline

- Buy
- First Launch
- Start Level 2

App Binary

- Downloading
- Playing
- Downloading
- Playing
- Downloading
- Playing
Launch Timeline

App Binary

Time

Buy

First Launch

Start Level 2

Start Level 3

Purchase Item

Downloading

Playing

Downloading

Playing

Downloading

Playing

Downloading

Playing
Launch Timeline
Launch Timeline

- **Buy**
- **First Launch**
- **Run Level 2**
- **Run Level 3**
- **Purchase Item**

- **App Binary**
  - Initial Install Tags
  - Prefetched Tags

- **Time**
  - Downloading
  - Playing
  - Downloading
  - Playing
  - Downloading
  - Playing
Launch Timeline

- **App Binary**
- **Initial Install Tags**
- **Prefetched Tags**
- **Predictive Download**

Time:
- **Downloading**
- **First Launch**
- **Run Level 2**
- **Run Level 3**
- **Purchase Item**
- **Playing**
Optimizing First Launch
Initial install and prefetched resources

Initial Install Tags

• Downloaded and installed with the app
• Up to 2GB
• Part of the “Size” shown in the App Store
Initial install and prefetched resources

Initial Install Tags
• Downloaded and installed with the app
• Up to 2GB
• Part of the “Size” shown in the App Store

Prefetched tag order
• Automatically prefetches after app download
• Up to 4GB – sizeof(Initial Install Tags)
• Follows order specified in Xcode
## Optimizing First Launch

Initial install and prefetched resources

<table>
<thead>
<tr>
<th>Tag Order Type</th>
<th>Total Size (MB)</th>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Initial Install Tags</strong></td>
<td>202</td>
<td>Level1Enemies (48 MB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Level1Map (65 MB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tutorial (89 MB)</td>
</tr>
<tr>
<td><strong>Prefetched Tag Order</strong></td>
<td>125</td>
<td>Level2Map (65 MB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Level2Enemies (60 MB)</td>
</tr>
<tr>
<td><strong>Download Only On Demand</strong></td>
<td>203</td>
<td>HolidayTheme (57 MB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Level3Enemies (59 MB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Level3Map (65 MB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SpecialWeapon (22 MB)</td>
</tr>
</tbody>
</table>
## Optimizing First Launch

Initial install and prefetched resources

<table>
<thead>
<tr>
<th>Tag Type</th>
<th>Tag Name</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Initial Install Tags</strong></td>
<td>Level1Enemies</td>
<td>48 MB</td>
</tr>
<tr>
<td></td>
<td>Level1Map</td>
<td>65 MB</td>
</tr>
<tr>
<td></td>
<td>Tutorial</td>
<td>89 MB</td>
</tr>
<tr>
<td><strong>Prefetched Tag Order</strong></td>
<td>Level2Map</td>
<td>65 MB</td>
</tr>
<tr>
<td></td>
<td>Level2Enemies</td>
<td>60 MB</td>
</tr>
<tr>
<td><strong>Download Only On Demand</strong></td>
<td>HolidayTheme</td>
<td>57 MB</td>
</tr>
<tr>
<td></td>
<td>Level3Enemies</td>
<td>59 MB</td>
</tr>
<tr>
<td></td>
<td>Level3Map</td>
<td>65 MB</td>
</tr>
<tr>
<td></td>
<td>SpecialWeapon</td>
<td>22 MB</td>
</tr>
</tbody>
</table>
## Optimizing First Launch

Initial install and prefetched resources

<table>
<thead>
<tr>
<th>Resource</th>
<th>Size (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Install Tags</td>
<td>202 MB</td>
</tr>
<tr>
<td>Level1Enemies</td>
<td>48 MB</td>
</tr>
<tr>
<td>Level1Map</td>
<td>65 MB</td>
</tr>
<tr>
<td>Tutorial</td>
<td>89 MB</td>
</tr>
<tr>
<td>Prefetched Tag Order</td>
<td>125 MB</td>
</tr>
<tr>
<td>Level2Map</td>
<td>65 MB</td>
</tr>
<tr>
<td>Level2Enemies</td>
<td>60 MB</td>
</tr>
<tr>
<td>Download Only On Demand</td>
<td>203 MB</td>
</tr>
<tr>
<td>HolidayTheme</td>
<td>57 MB</td>
</tr>
<tr>
<td>Level3Enemies</td>
<td>59 MB</td>
</tr>
<tr>
<td>Level3Map</td>
<td>65 MB</td>
</tr>
<tr>
<td>SpecialWeapon</td>
<td>22 MB</td>
</tr>
</tbody>
</table>
## Optimizing First Launch

Initial install and prefetched resources

<table>
<thead>
<tr>
<th>Tag Category</th>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Initial Install Tags</strong></td>
<td>(202 MB)</td>
</tr>
<tr>
<td>Level1Enemies</td>
<td>(48 MB)</td>
</tr>
<tr>
<td>Level1Map</td>
<td>(65 MB)</td>
</tr>
<tr>
<td>Tutorial</td>
<td>(89 MB)</td>
</tr>
<tr>
<td><strong>Prefetched Tag Order</strong></td>
<td>(125 MB)</td>
</tr>
<tr>
<td>Level2Map</td>
<td>(65 MB)</td>
</tr>
<tr>
<td>Level2Enemies</td>
<td>(60 MB)</td>
</tr>
<tr>
<td><strong>Download Only On Demand</strong></td>
<td>(203 MB)</td>
</tr>
<tr>
<td>HolidayTheme</td>
<td>(57 MB)</td>
</tr>
<tr>
<td>Level3Enemies</td>
<td>(59 MB)</td>
</tr>
<tr>
<td>Level3Map</td>
<td>(65 MB)</td>
</tr>
<tr>
<td>SpecialWeapon</td>
<td>(22 MB)</td>
</tr>
</tbody>
</table>
Predictive Loading
Launch Timeline

App Binary

Initial Install Tags

Prefetched Tags

Time

Buy

First Launch

Run Level 2

Run Level 3

Purchase Item

Downloading

Playing

Downloaded

Playing

Downloaded

Playing
Linear Access Pattern

Majority of assets will be used
Tag size isn’t that critical
• Access tags early
Linear Access Pattern

Majority of assets will be used

Tag size isn’t that critical

• Access tags early
Random Access Pattern

Access order is indeterminate
Use many tags
- Tag small groups of assets for progressive download and consumption
- Download sets of tags proactively
- End accessing does not mean deletion
Random Access Pattern

Access order is indeterminate

Use many tags

• Tag small groups of assets for progressive download and consumption

• Download sets of tags proactively

• End accessing does not mean deletion
Explorative Access Pattern

Limited Prediction
- Many possibilities will not be used
Use many tags
- Load subset of possible resources
- Use hints to narrow the choices
- Quickly end accessing on unused resource requests
Explorative Access Pattern

Limited Prediction
• Many possibilities will not be used
Use many tags
• Load subset of possible resources
• Use hints to narrow the choices
• Quickly end accessing on unused resource requests
Implementation Details
## On-Demand Resources

<table>
<thead>
<tr>
<th>Resource</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS App Bundle</td>
<td>Up to 4GB</td>
</tr>
<tr>
<td>tvOS App Bundle</td>
<td>Up to 200MB</td>
</tr>
<tr>
<td>On-Demand Resources</td>
<td>Up to 20GB</td>
</tr>
<tr>
<td>Initial Install Tags</td>
<td>Up to 2GB</td>
</tr>
<tr>
<td>Prefetch Tags</td>
<td>Up to 4GB – (Initial Install Tags)</td>
</tr>
</tbody>
</table>
## On-Demand Resources

<table>
<thead>
<tr>
<th>Resource</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS App Bundle</td>
<td>Up to 4GB</td>
</tr>
<tr>
<td>tvOS App Bundle</td>
<td>Up to 200MB</td>
</tr>
<tr>
<td>On-Demand Resources</td>
<td>Up to 20GB</td>
</tr>
<tr>
<td>Initial Install Tags</td>
<td>Up to 2GB</td>
</tr>
<tr>
<td>Prefetch Tags</td>
<td>Up to 4GB – (Initial Install Tags)</td>
</tr>
<tr>
<td>Active Resources</td>
<td>Up to 2GB</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Max Size</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>iOS App Bundle</td>
<td>Up to 4GB</td>
</tr>
<tr>
<td>tvOS App Bundle</td>
<td>Up to 200MB</td>
</tr>
<tr>
<td>On-Demand Resources</td>
<td>Up to 20GB</td>
</tr>
<tr>
<td>Initial Install Tags</td>
<td>Up to 2GB</td>
</tr>
<tr>
<td>Prefetch Tags</td>
<td>Up to 4GB – (Initial Install Tags)</td>
</tr>
<tr>
<td>Active Resources</td>
<td>Up to 2GB</td>
</tr>
<tr>
<td>One Tag</td>
<td>Up to 512MB (64MB recommended)</td>
</tr>
<tr>
<td>Resource</td>
<td>Size Limit</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>iOS App Bundle</td>
<td>Up to 4GB</td>
</tr>
<tr>
<td>tvOS App Bundle</td>
<td>Up to 200MB</td>
</tr>
<tr>
<td>On-Demand Resources</td>
<td>Up to 20GB</td>
</tr>
<tr>
<td>Initial Install Tags</td>
<td>Up to 2GB</td>
</tr>
<tr>
<td>Prefetch Tags</td>
<td>Up to 4GB – (Initial Install Tags)</td>
</tr>
<tr>
<td>Active Resources</td>
<td>Up to 2GB</td>
</tr>
<tr>
<td>One Tag</td>
<td>Up to 512MB (64MB recommended)</td>
</tr>
<tr>
<td>Total Asset Packs</td>
<td>Up to 1000</td>
</tr>
</tbody>
</table>
Asset Packs

GreatRPG.app

App Binary

“Level01NPCs”

“Level01Enemies”

“Level02NPCs”

“Level02Enemies”
Asset Packs

GreatRPG.app

App Binary

“Level01NPCs”

“Level01Enemies”

“Level02NPCs”

“Level02Enemies”
Asset Packs

GreatRPG.app

App Binary

“Level01NPCs”

“Level01Enemies”

“Level02NPCs”

“Level02Enemies”
Asset Packs

GreatGame.app

- "Level01NPCs"
- "Level01Enemies"
- "Level02NPCs"
- "Level02Enemies"
- "Level01NPCs","Level02NPCs"
- "Level01Enemies","Level02NPCs"
Asset Packs
GreatGame.app

Four tags

“Level01NPCs”
“Level01Enemies”
“Level02NPCs”
“Level02Enemies”
“Level01NPCs”, “Level02NPCs”
“Level01Enemies”, “Level02NPCs”
Asset Packs
GreatGame.app

Four tags

Six asset packs

“Level01NPCs”
“Level01Enemies”
“Level02NPCs”
“Level02Enemies”
“Level01NPCs,” “Level02NPCs”
“Level01Enemies,” “Level02NPCs”
Optimizing ODR App Updates
ODR App Updates
V1.0
ODR App Updates
1.0 -> 2.0
ODR App Updates

Overview

App Binary

2.0
ODR App Updates

Overview

Updated resources

• Redownload only when accessed
ODR App Updates

Overview

Updated resources
• Redownload only when accessed

Unchanged resources
• Usually remain on device
• Can be accessed without redownload
ODR App Updates

Overview

Updated resources
• Redownload only when accessed

Unchanged resources
• Usually remain on device
• Can be accessed without redownload

New resources
• Download only when accessed
ODR App Updates
Best practices

Avoid making unnecessary modifications to tagged resources
  - One changed resource triggers redownload of the asset pack
  - Consider addendum tags; “Level 01” and “Level 01 Update 1”
ODR App Updates

Best practices

Avoid making unnecessary modifications to tagged resources

• One changed resource triggers redownload of the asset pack
• Consider addendum tags; “Level 01” and “Level 01 Update 1”

Keep tags consistent from version to version
ODR App Updates

Best practices

Avoid making unnecessary modifications to tagged resources

- One changed resource triggers redownload of the asset pack
- Consider addendum tags; “Level 01” and “Level 01 Update 1”

Keep tags consistent from version to version

Design with separation of updatable content vs. static content
Intelligent Content Caching

How it works
Intelligent Content Caching

How it works

Resources may be purged when the system demands disk space

• Ending access does not mean deletion
Intelligent Content Caching

How it works

Resources may be purged when the system demands disk space
• Ending access does not mean deletion

Variables that inform purge order
• Last used timestamp
• Preservation priority
  – Isolated to your application
• Application-running state
Intelligent Content Caching

How it works

Resources may be purged when the system demands disk space

- Ending access does not mean deletion

Variables that inform purge order

- Last used timestamp
- Preservation priority
  - Isolated to your application
- Application-running state

Don’t use tmp or caches

- Purged first. Purged completely.
Conclusion
Use On-Demand Resources
On-Demand Resources

Smaller main app bundle
• Faster initial download
On-Demand Resources

Smaller main app bundle
• Faster initial download

Richer app content
• Up to 20GB available on-demand
On-Demand Resources

Smaller main app bundle
• Faster initial download

Richer app content
• Up to 20GB available on-demand

More apps installed and ready to run
• Reduces need to manage storage
More Information

https://developer.apple.com/wwdc16/221
## Related Sessions

<table>
<thead>
<tr>
<th>Session</th>
<th>Location</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>What’s New in tvOS</td>
<td>Presidio</td>
<td>Tuesday 3:00PM</td>
</tr>
<tr>
<td>Designing for tvOS</td>
<td>Presidio</td>
<td>Tuesday 4:00PM</td>
</tr>
<tr>
<td>Controlling Game Input for Apple TV</td>
<td>Mission</td>
<td>Wednesday 5:00PM</td>
</tr>
<tr>
<td>Lab</td>
<td>On-Demand Resources Lab</td>
<td>Frameworks Lab B</td>
</tr>
</tbody>
</table>