What’s New in Game Center

Session 504

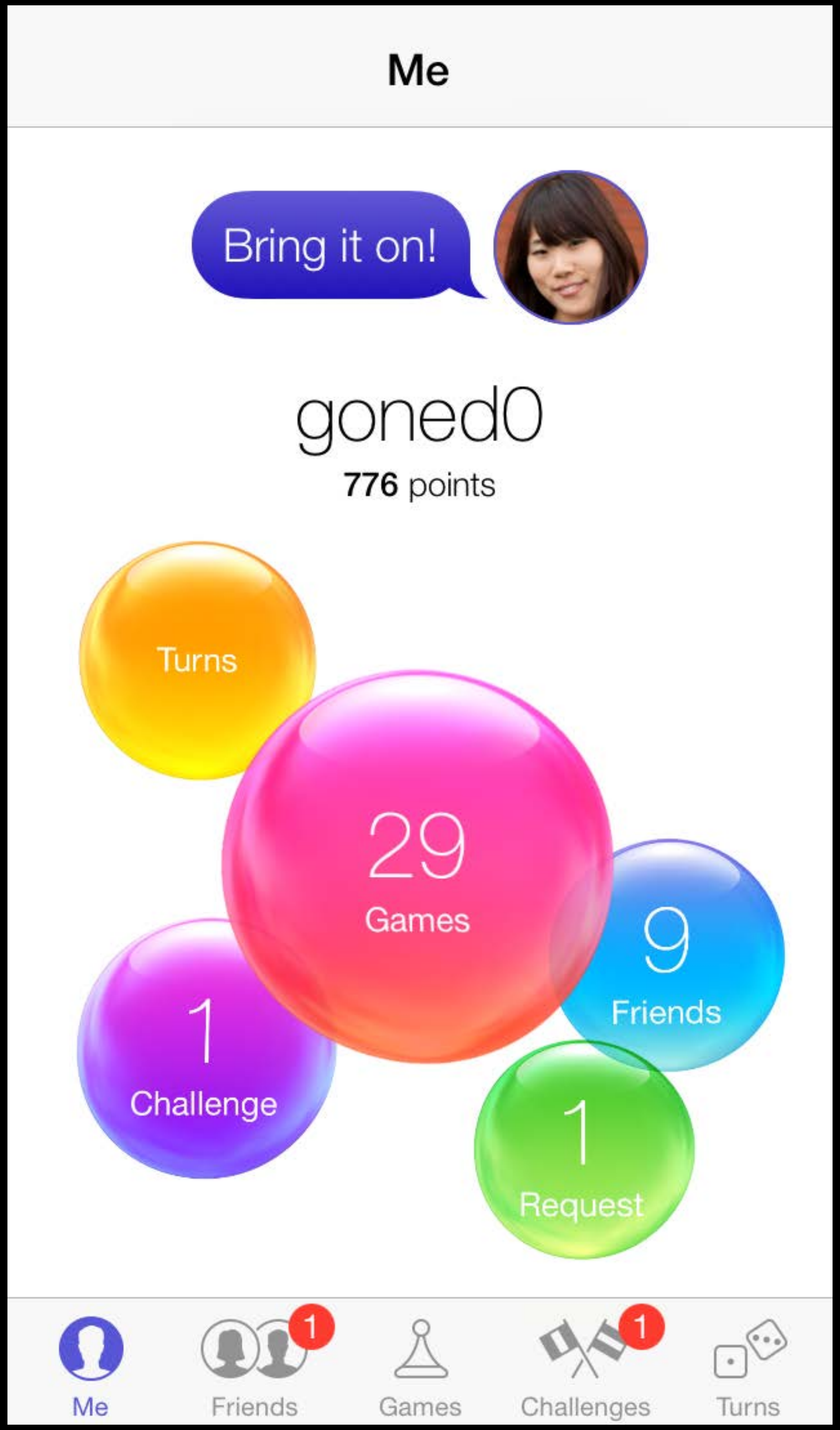
Dan Kurtz
iOS Engineer

These are confidential sessions—please refrain from streaming, blogging, or taking pictures

What Is Game Center?

- Track and share scores and achievements
- Apple-hosted, turn-based multiplayer
- Multiplayer matchmaking
- Real-time multiplayer support
- Available on iOS and OS X

NEW



Me

Bring it on!

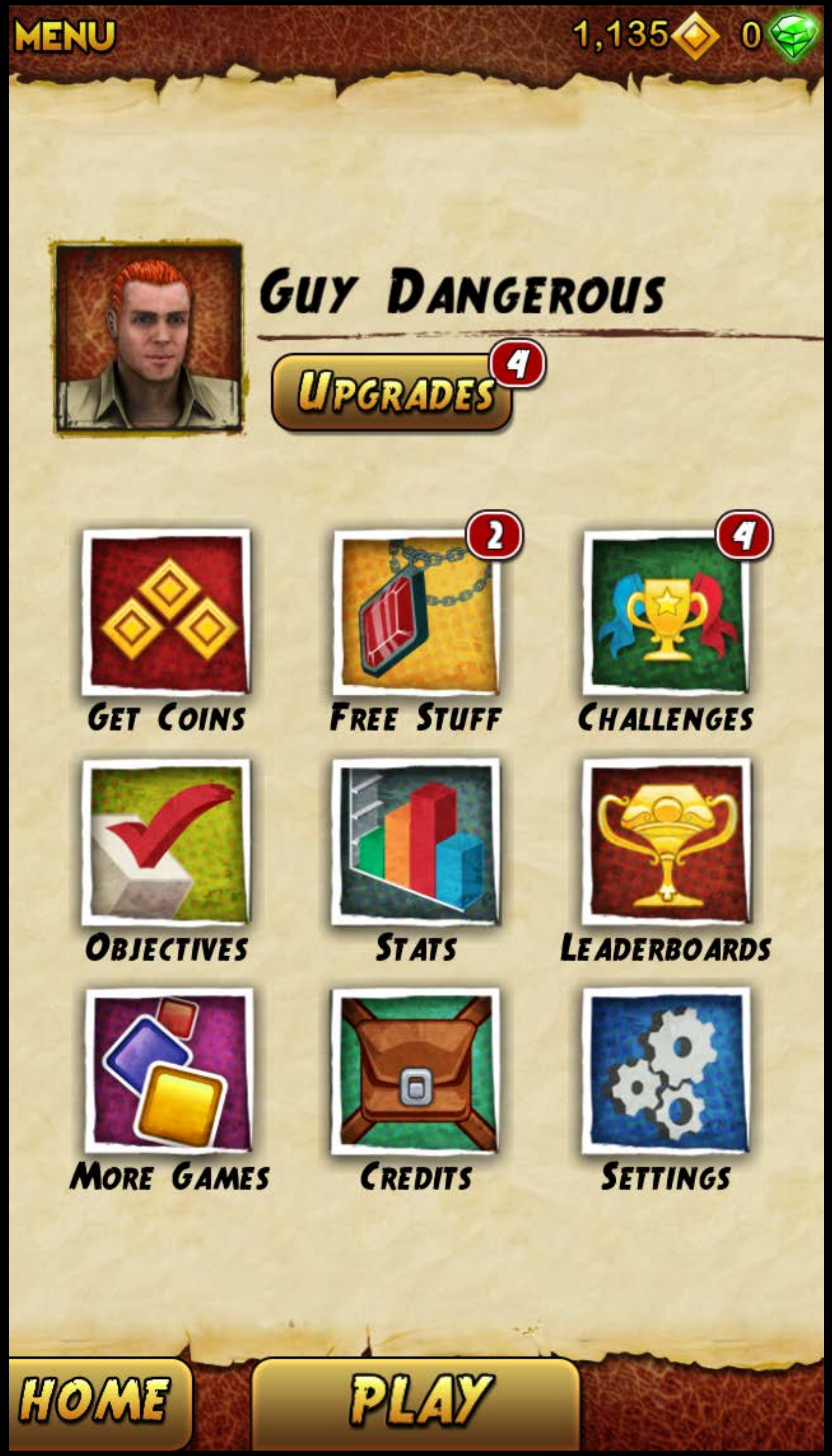
goned0

776 points

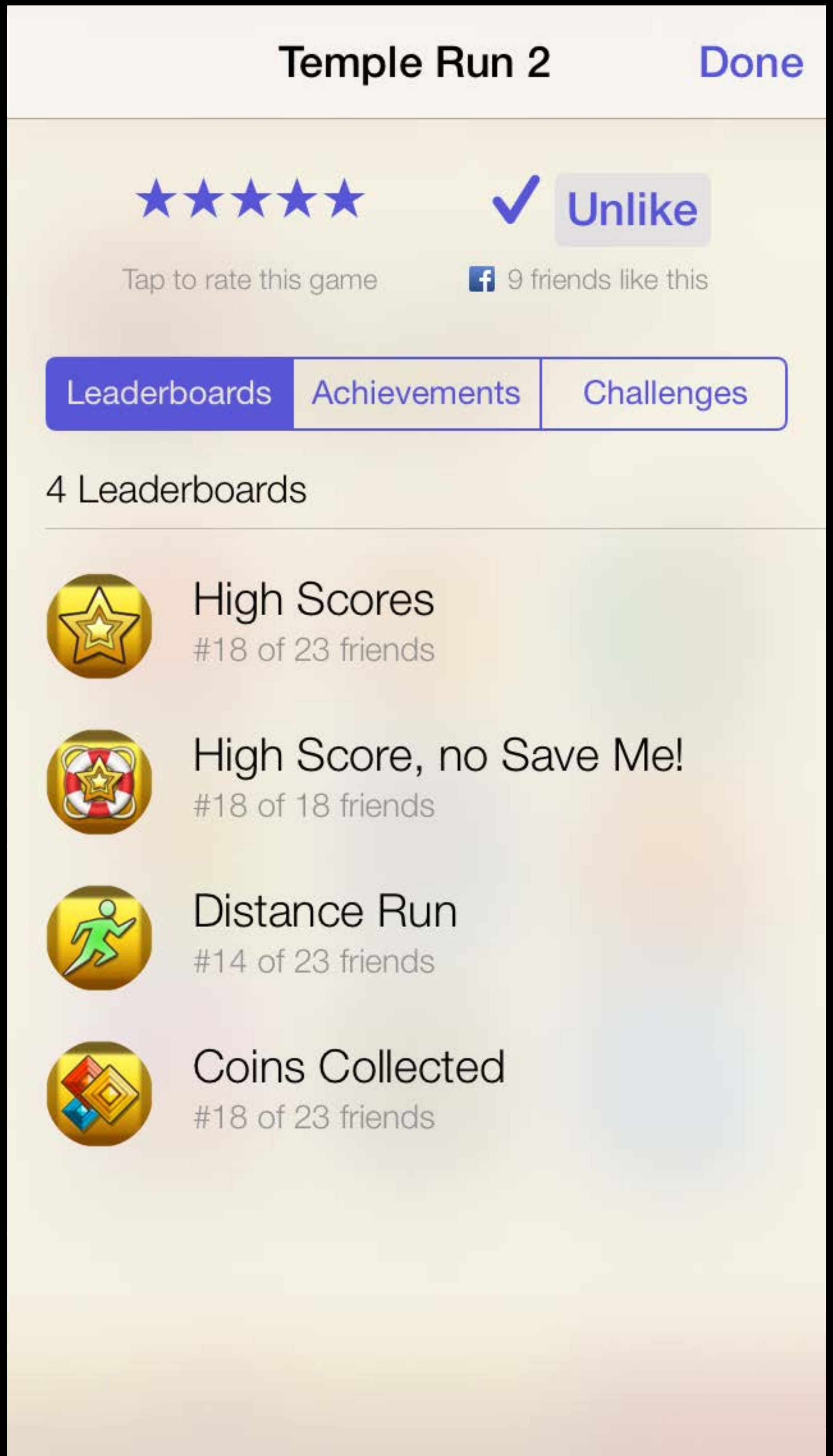
What You Will Learn

- Game Center In-Game
- Leaderboard Sets
- Most Recent Score Leaderboards
- Improved Security
- Challenges
- Fundamental API Changes

NEW


MENU
1,135
0
GUY DANGEROUS
UPGRADES
4
GET COINS
FREE STUFF
2
CHALLENGES
4
OBJECTIVES
STATS
LEADERBOARDS
MORE GAMES
CREDITS
SETTINGS
HOME
PLAY

NEW



Temple Run 2

4 Leaderboards

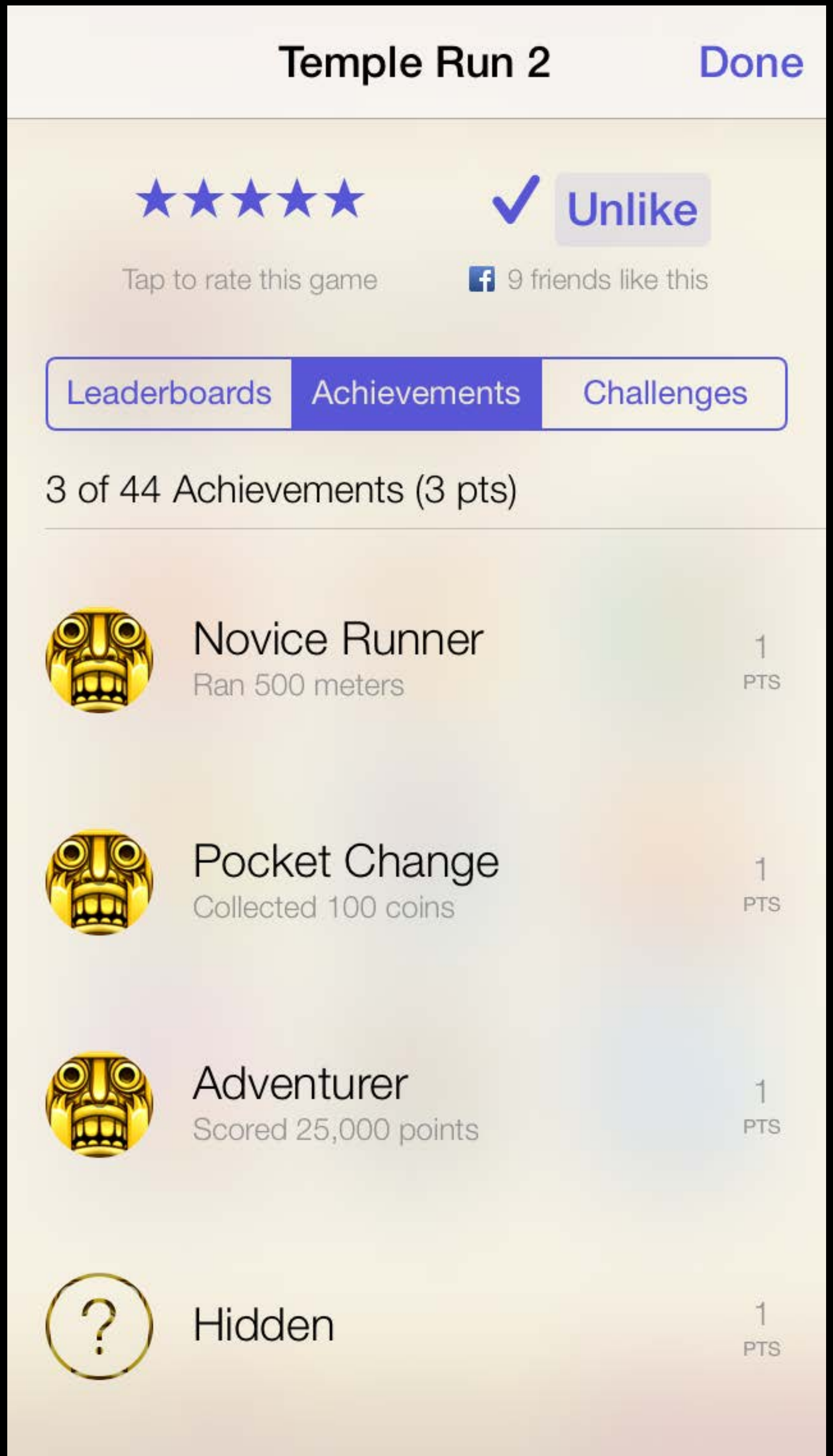
High Scores
#18 of 23 friends

High Score, no Save Me!
#18 of 18 friends

Distance Run
#14 of 23 friends

Coins Collected
#18 of 23 friends

NEW



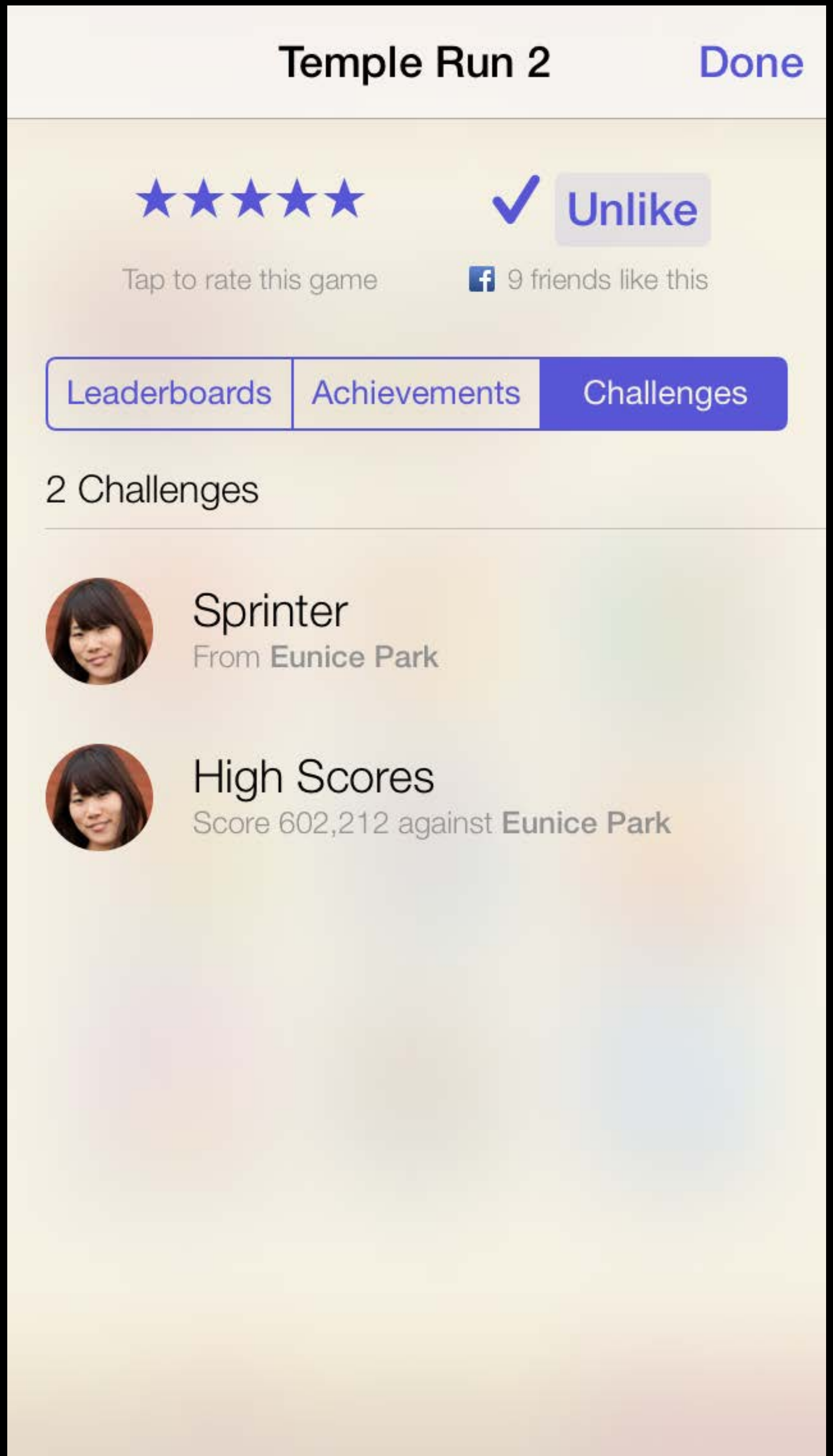
Tap to rate this game

9 friends like this

3 of 44 Achievements (3 pts)

Novice Runner	Ran 500 meters	1 PTS
Pocket Change	Collected 100 coins	1 PTS
Adventurer	Scored 25,000 points	1 PTS
Hidden		1 PTS

NEW



Temple Run 2 Done

★★★★★

Tap to rate this game

✓ Unlike

9 friends like this

Leaderboards | Achievements | Challenges

2 Challenges

Sprinter
From **Eunice Park**

High Scores
Score 602,212 against **Eunice Park**

GKGameCenterViewController

```
@implementation MyViewController

- (void)showUIButtonPressed: (id)sender {
    // get the shared VC instance
    GKGameCenterViewController *gcvc = [GKGameCenterViewController
        sharedController];
    gcvc.gameCenterDelegate = self;

    // set the starting view state
    gcvc.viewState = GKGameCenterViewControllerStateDefault;

    // throttle or pause any rendering for performance
    [self throttleRendering];

    [self presentViewController: gcvc animated: YES completion: nil];
}
```

GKGameCenterViewController

```
@implementation MyViewController

- (void)showUIButtonPressed: (id)sender {
    // get the shared VC instance
    GKGameCenterViewController *gcvc = [GKGameCenterViewController
        sharedController];
    gcvc.gameCenterDelegate = self;

    // set the starting view state
    gcvc.viewState = GKGameCenterViewControllerStateDefault;

    // throttle or pause any rendering for performance
    [self throttleRendering];

    [self presentViewController: gcvc animated: YES completion: nil];
}
```

GKGameCenterViewController

```
@implementation MyViewController

- (void)showUIButtonPressed: (id)sender {
    // get the shared VC instance
    GKGameCenterViewController *gcvc = [GKGameCenterViewController
        sharedController];
    gcvc.gameCenterDelegate = self;

    // set the starting view state
    gcvc.viewState = GKGameCenterViewControllerStateDefault;

    // throttle or pause any rendering for performance
    [self throttleRendering];

    [self presentViewController: gcvc animated: YES completion: nil];
}
```

GKGameCenterViewController

```
@implementation MyViewController

- (void)showUIButtonPressed: (id)sender {
    // get the shared VC instance
    GKGameCenterViewController *gcvc = [GKGameCenterViewController
        sharedController];
    gcvc.gameCenterDelegate = self;

    // set the starting view state
    gcvc.viewState = GKGameCenterViewControllerStateDefault;

    // throttle or pause any rendering for performance
    [self throttleRendering];

    [self presentViewController: gcvc animated: YES completion: nil];
}
```

GKGameCenterViewController

```
@implementation MyViewController

- (void)showUIButtonPressed: (id)sender {
    // get the shared VC instance
    GKGameCenterViewController *gcvc = [GKGameCenterViewController
        sharedController];
    gcvc.gameCenterDelegate = self;

    // set the starting view state
    gcvc.viewState = GKGameCenterViewControllerStateDefault;

    // throttle or pause any rendering for performance
    [self throttleRendering];

    [self presentViewController: gcvc animated: YES completion: nil];
}
```

GKGameCenterViewController

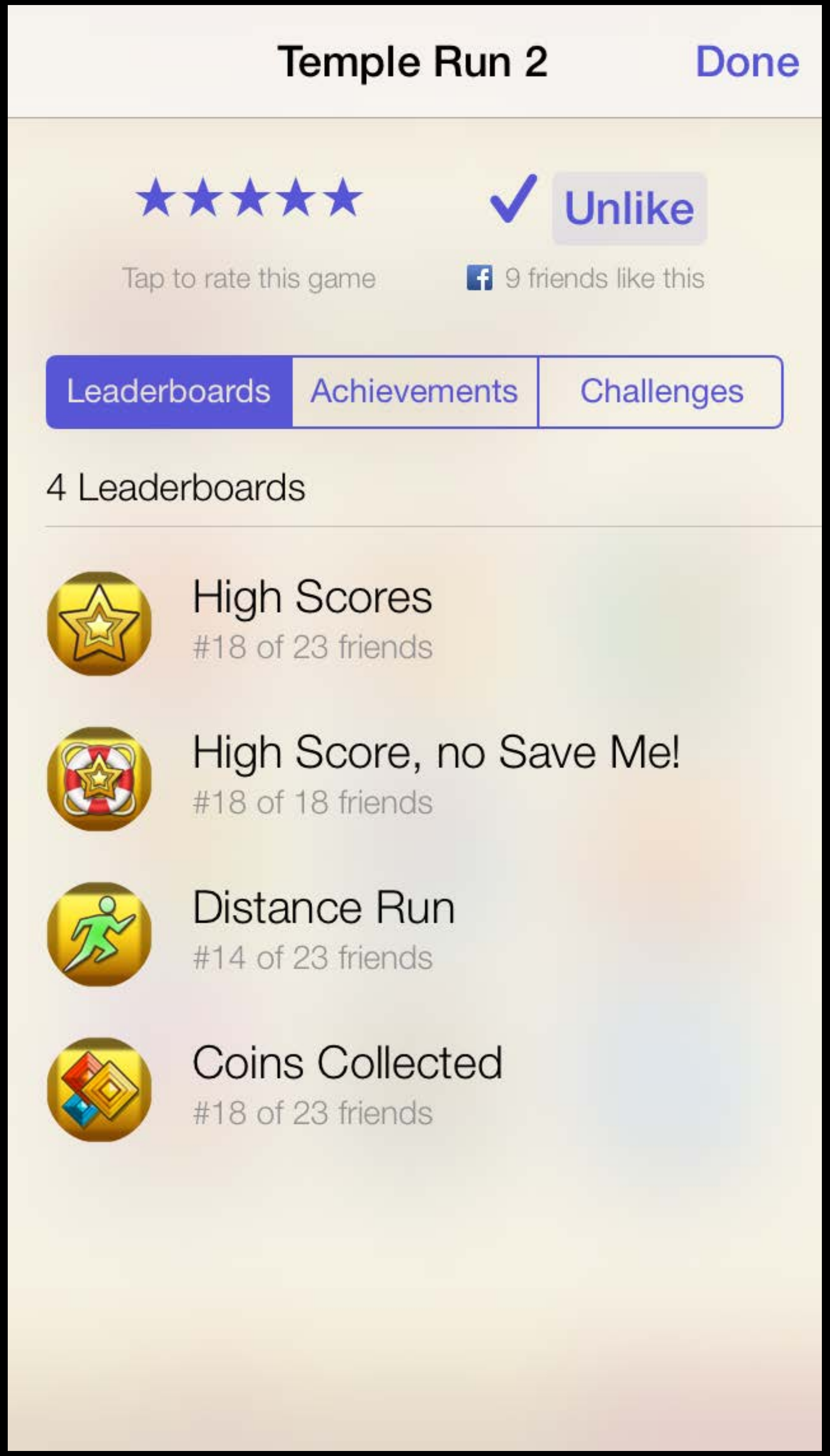
```
@implementation MyViewController

- (void)showUIButtonPressed: (id)sender {
    // get the shared VC instance
    GKGameCenterViewController *gcvc = [GKGameCenterViewController
        sharedController];
    gcvc.gameCenterDelegate = self;

    // set the starting view state
    gcvc.viewState = GKGameCenterViewControllerStateDefault;

    // throttle or pause any rendering for performance
    [self throttleRendering];

    [self presentViewController: gcvc animated: YES completion: nil];
}
```

GKGameCenterViewController

- Partially transparent
- Performance tip: Throttle rendering

GKGameCenterViewController

- Partially transparent
- Performance tip: Throttle rendering



Temple Run 2 Done

★★★★★ Tap to rate this game

✓ Unlike 9 friends like this

Leaderboards | Achievements | Challenges

4 Leaderboards

High Scores
#18 of 23 friends

High Score, no Save Me!
#18 of 18 friends

Distance Run
#14 of 23 friends

Coins Collected
#18 of 23 friends

Leaderboard Sets

Leaderboards

	Previous		
Max Leaderboards	25		
Organization	Flat		

Leaderboards

NEW

	Previous	Now
Max Leaderboards	25	100
Organization	Flat	Flat

Leaderboards

NEW

	Previous	Now	Now with Sets
Max Leaderboards	25	100	500
Organization	Flat	Flat	Leaderboard Sets
Max Sets			100
Leaderboards per Set			100

Leaderboard Sets

NEW

Water Planet
Boss Level
Medium

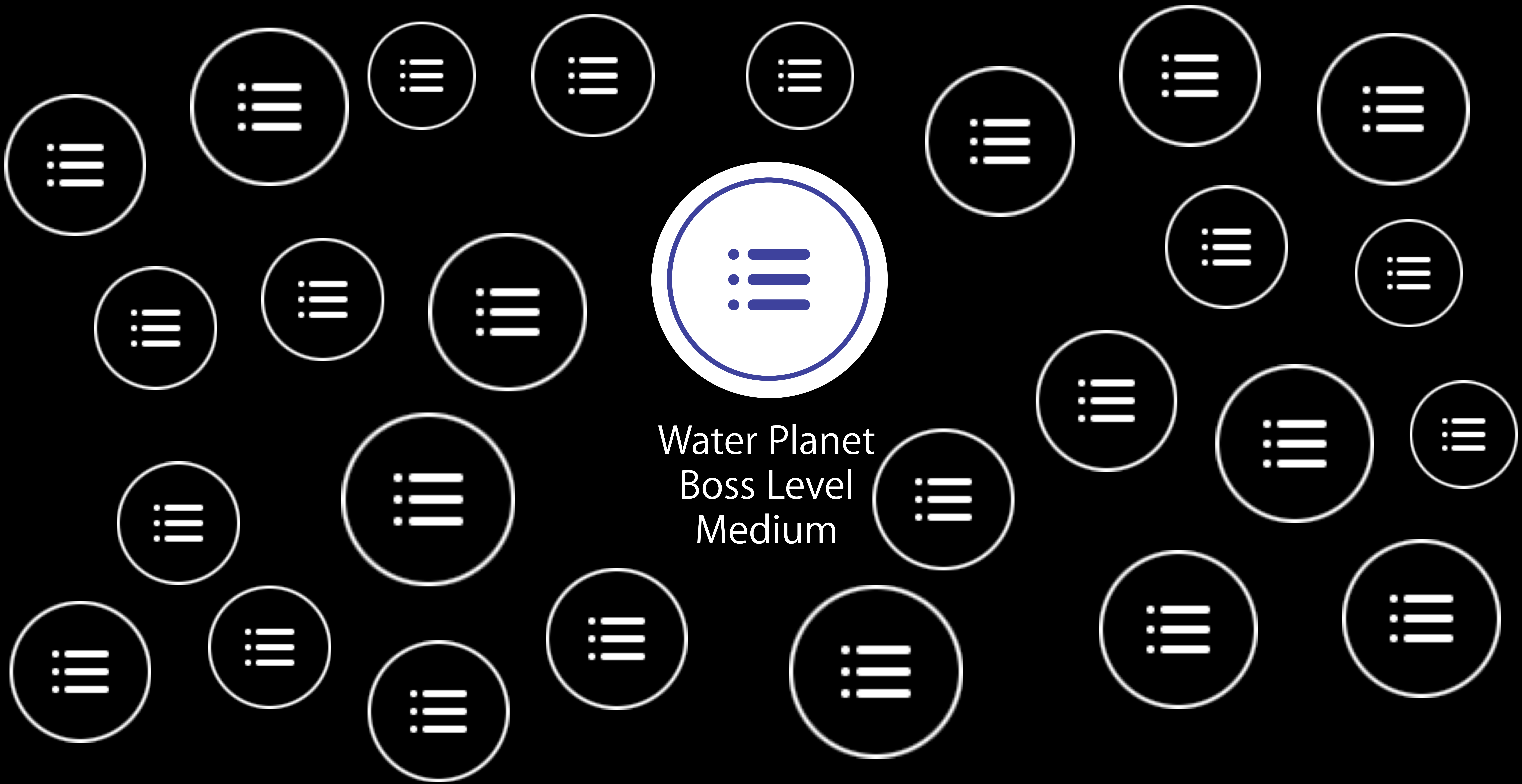
Leaderboard Sets

NEW

Water Planet
Boss Level
Medium

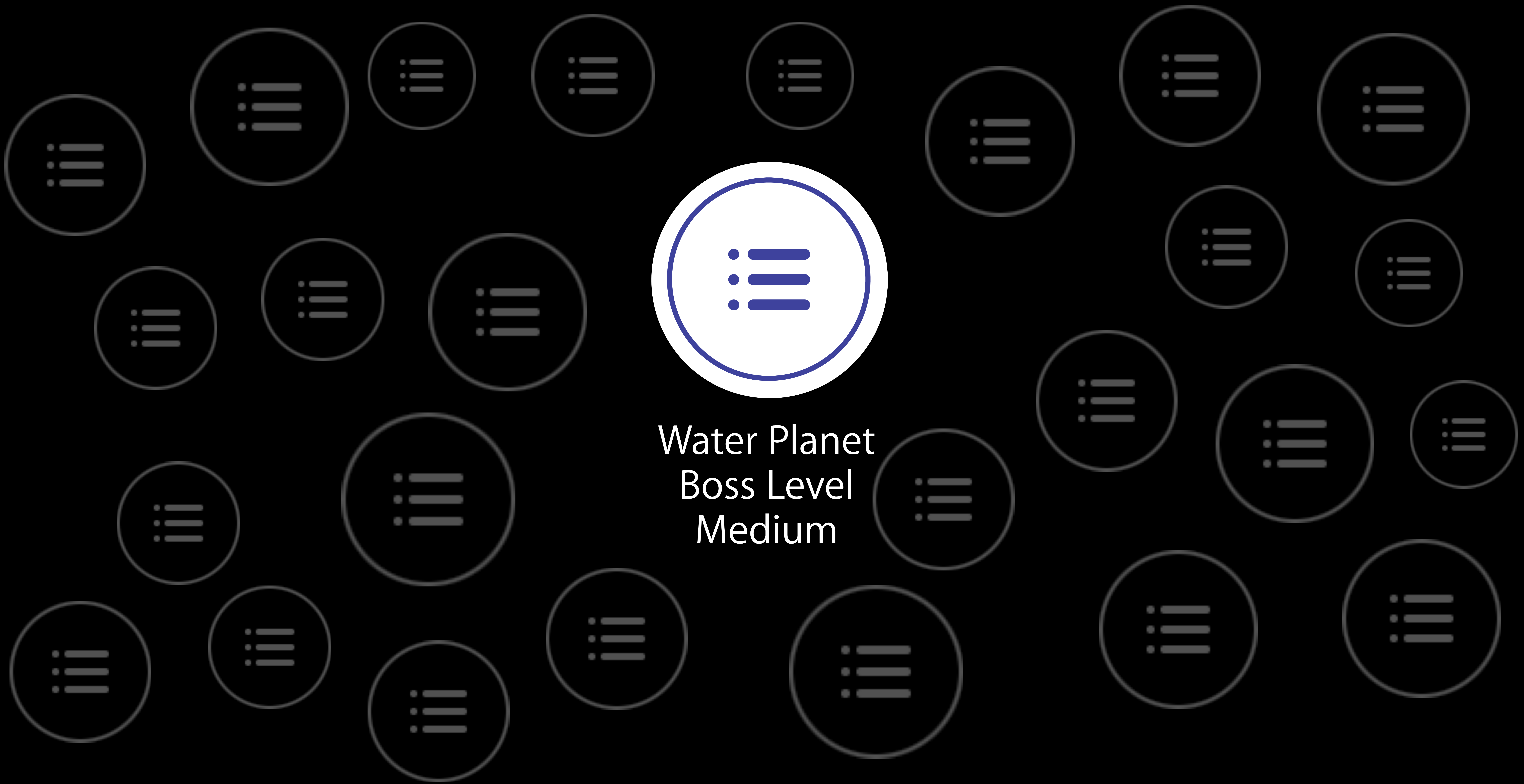
Leaderboard Sets

NEW

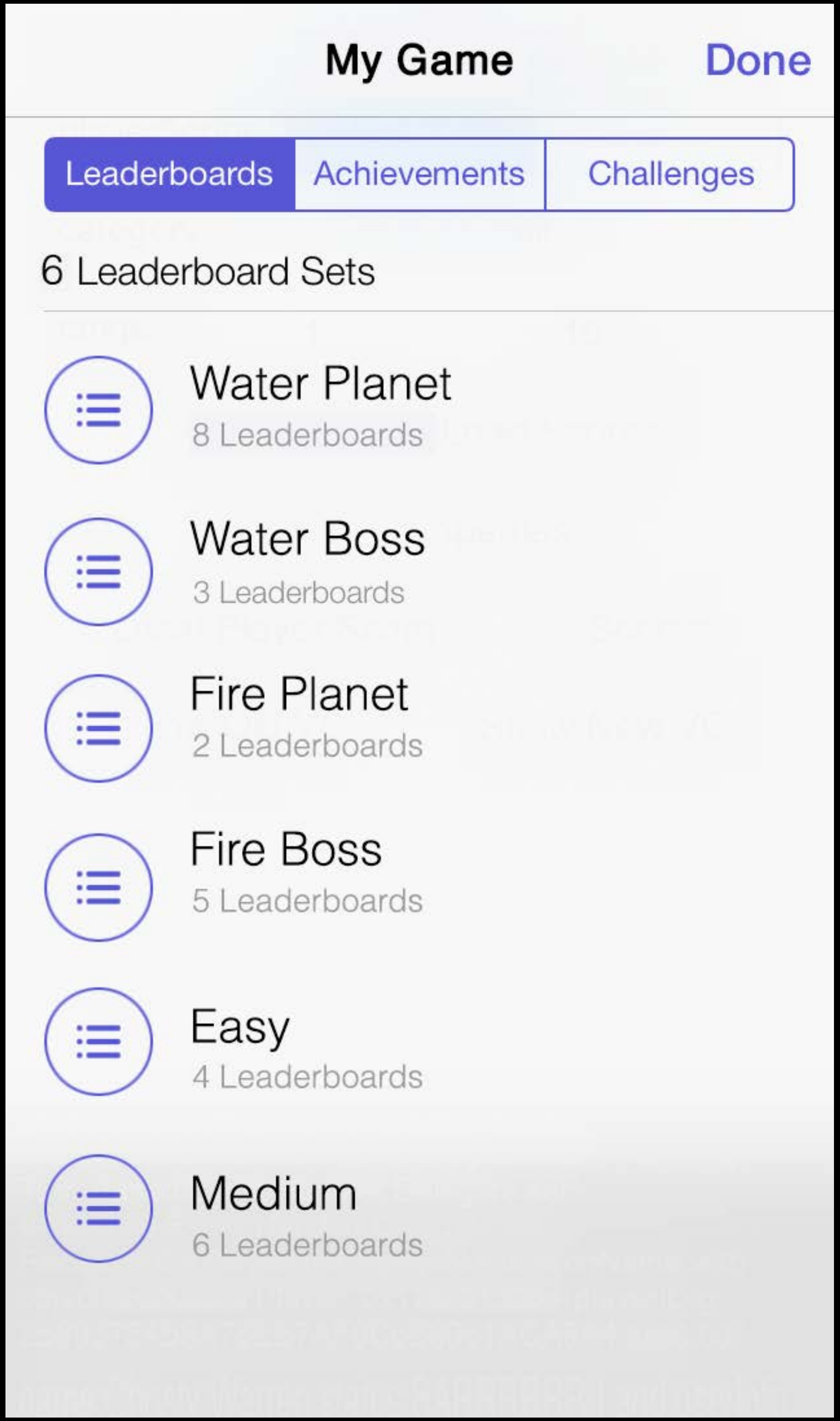


Leaderboard Sets

NEW



Water Planet
Boss Level
Medium



6 Leaderboard Sets

Water Planet
8 Leaderboards

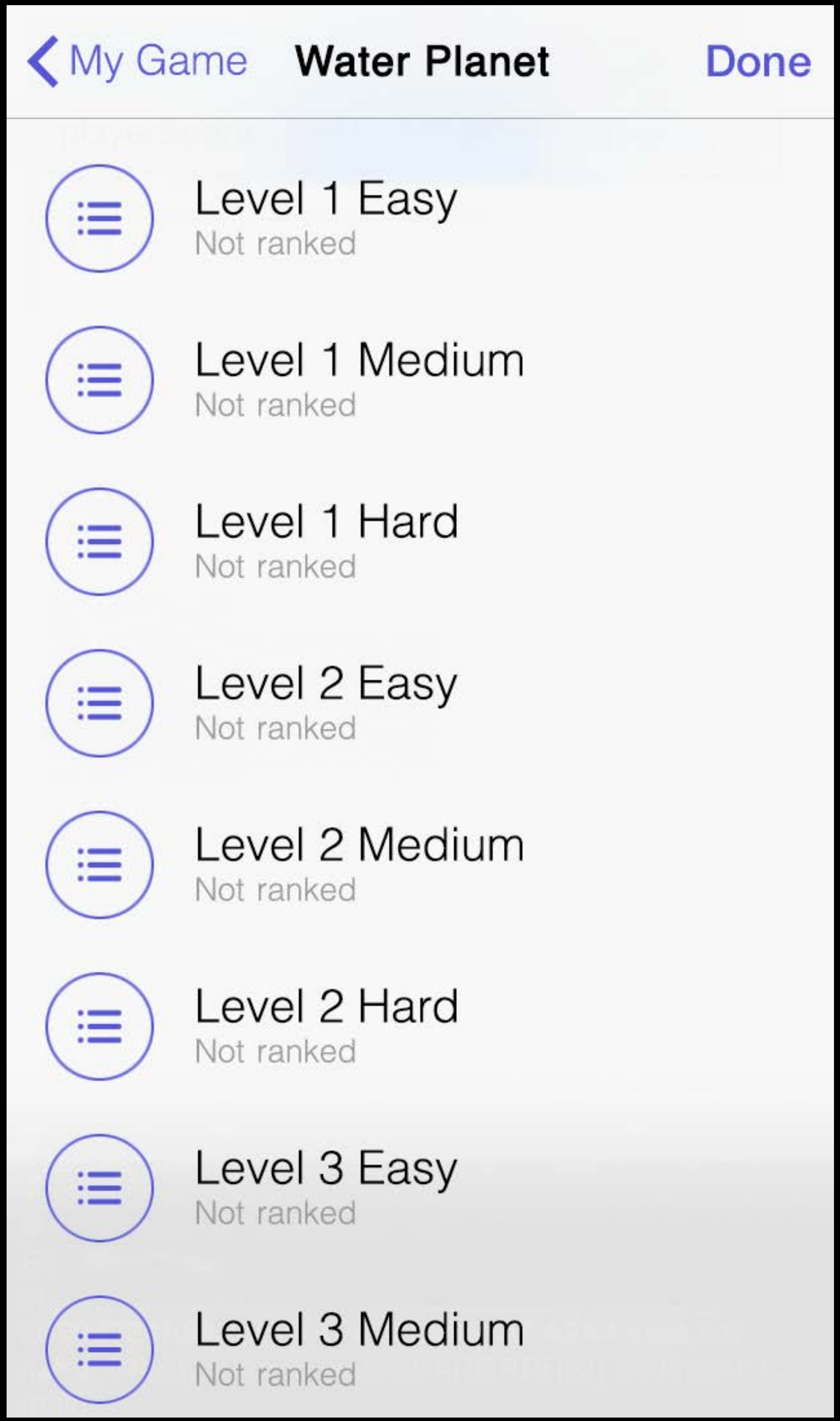
Water Boss
3 Leaderboards

Fire Planet
2 Leaderboards

Fire Boss
5 Leaderboards

Easy
4 Leaderboards

Medium
6 Leaderboards



Level 1 Easy
Not ranked

Level 1 Medium
Not ranked

Level 1 Hard
Not ranked

Level 2 Easy
Not ranked

Level 2 Medium
Not ranked

Level 2 Hard
Not ranked

Level 3 Easy
Not ranked

Level 3 Medium
Not ranked

Leaderboard Sets

Water Planet
Boss Level
Medium

Leaderboard Sets

Leaderboard Sets

Leaderboard Sets

- Water Planet

Leaderboard Sets

- Water Planet
- Boss Level for Planet

Leaderboard Sets

- Water Planet
- Boss Level for Planet
- Medium Difficulty

Leaderboard Sets

Water Planet Set

Water Boss Set

Medium Set

Leaderboard Sets

Water Planet Set

Water Boss Set

Medium Set

Leaderboard Sets

Water
Boss Med

Water Planet
Set

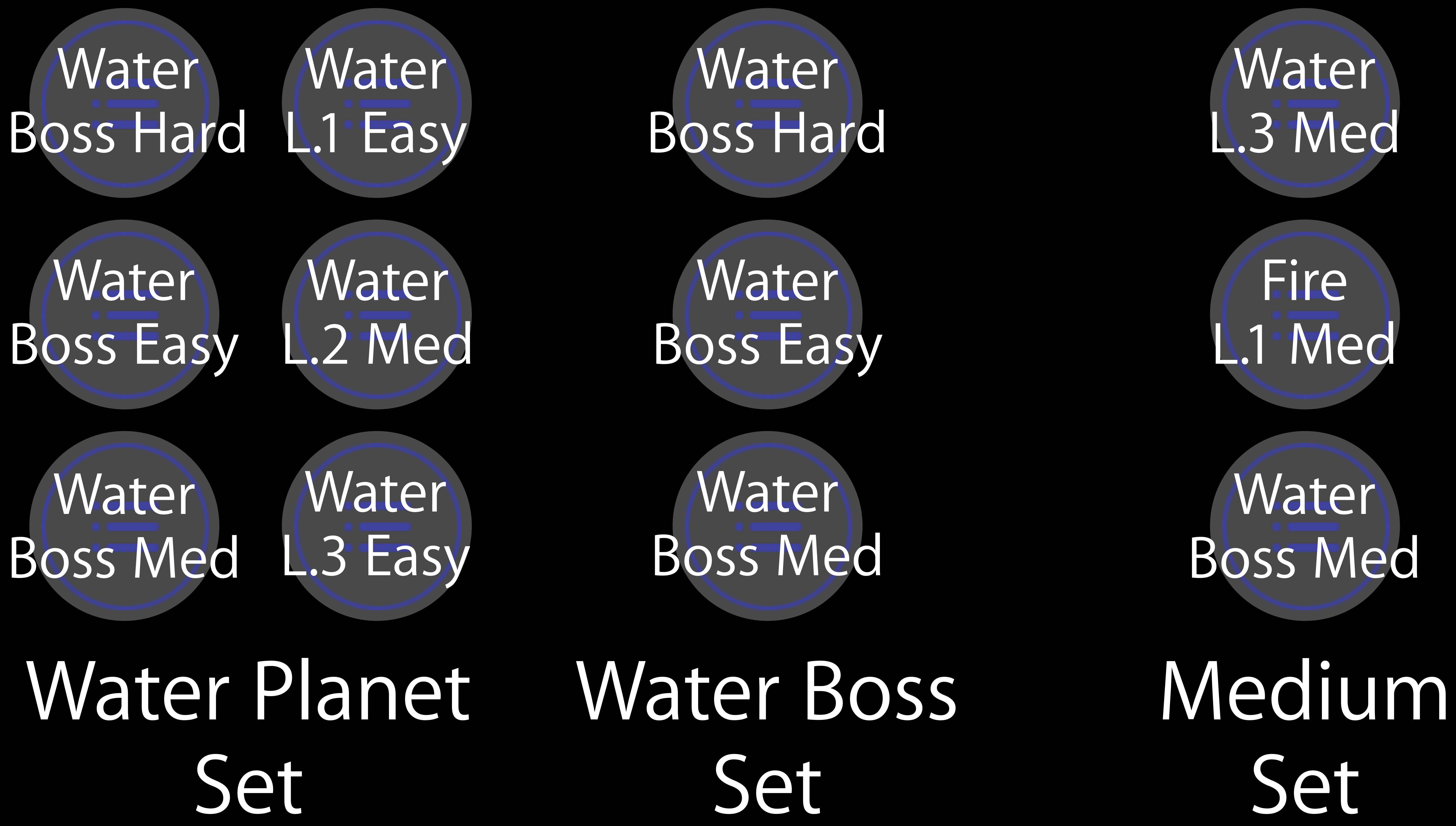
Water
Boss Med

Water Boss
Set

Water
Boss Med

Medium
Set

Leaderboard Sets



Water Boss Hard

Water L.1 Easy

Water Boss Easy

Water L.2 Med

Water Boss Med

Water L.3 Easy

Water Planet Set

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss Set

Water L.3 Med

Fire L.1 Med

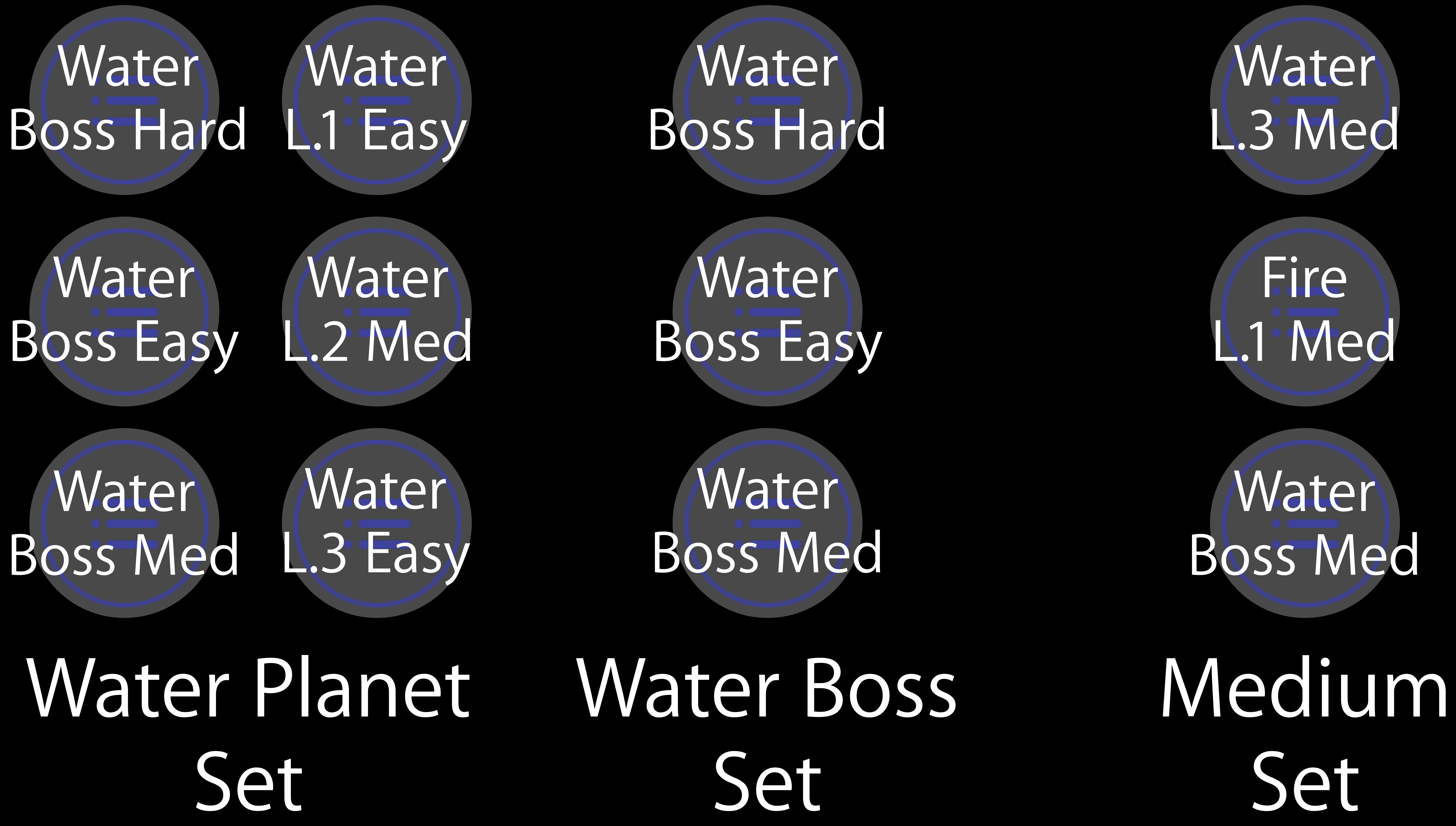
Water Boss Med

Medium Set

Naming Sets

- Sets have names (just like leaderboards)
- Leaderboard name can change depending on set

Naming Sets



Water Boss Hard

Water Boss Easy

Water Boss Med

Water L.1 Easy

Water L.2 Med

Water L.3 Easy

Water Planet Set

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss Set

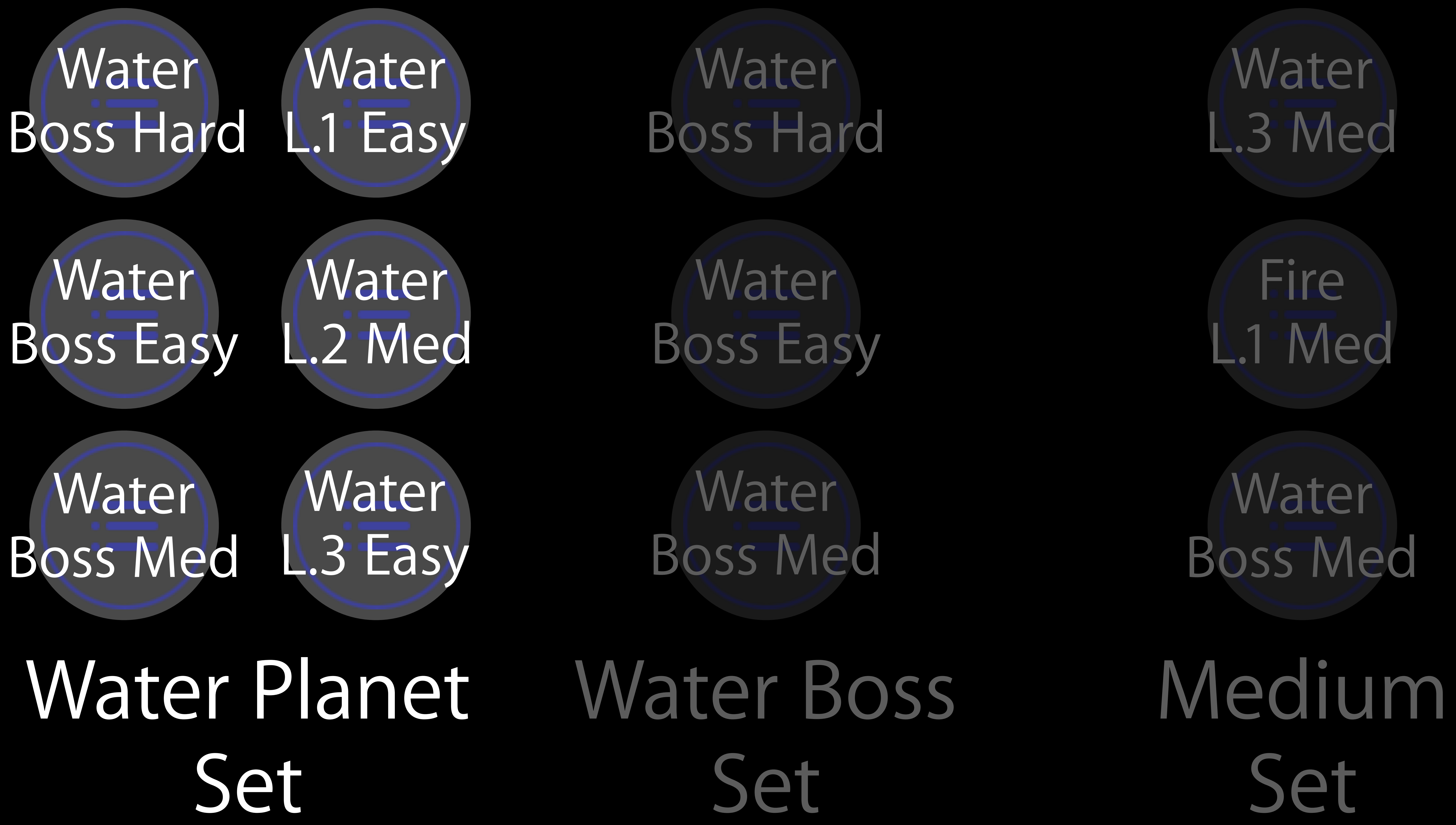
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Water Boss Hard

Water L.1 Easy

Water Boss Easy

Water L.2 Med

Water Boss Med

Water L.3 Easy

Water Planet Set

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss Set

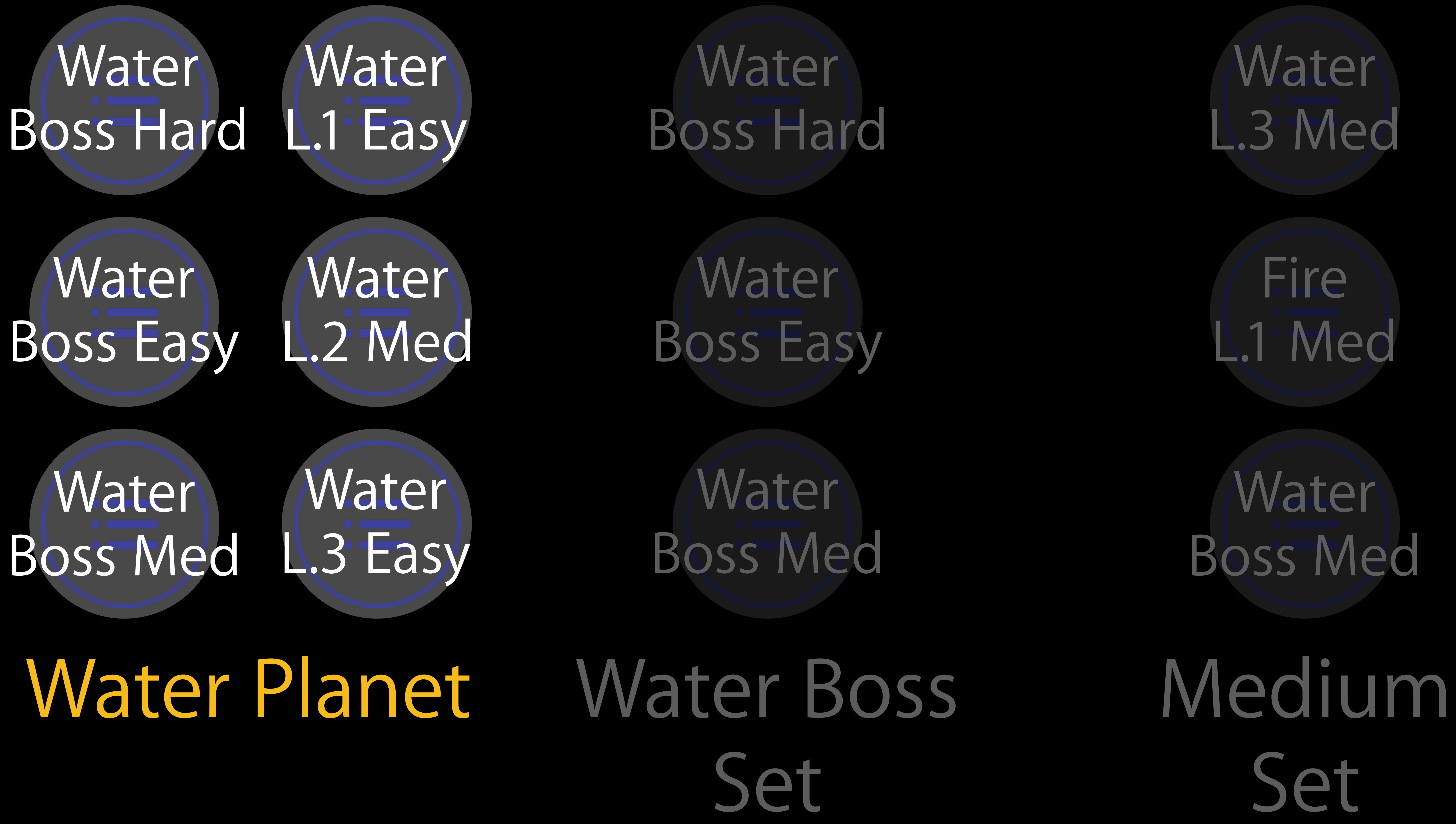
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Water Boss Hard

Water L.1 Easy

Water Boss Easy

Water L.2 Med

Water Boss Med

Water L.3 Easy

Water Planet

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss Set

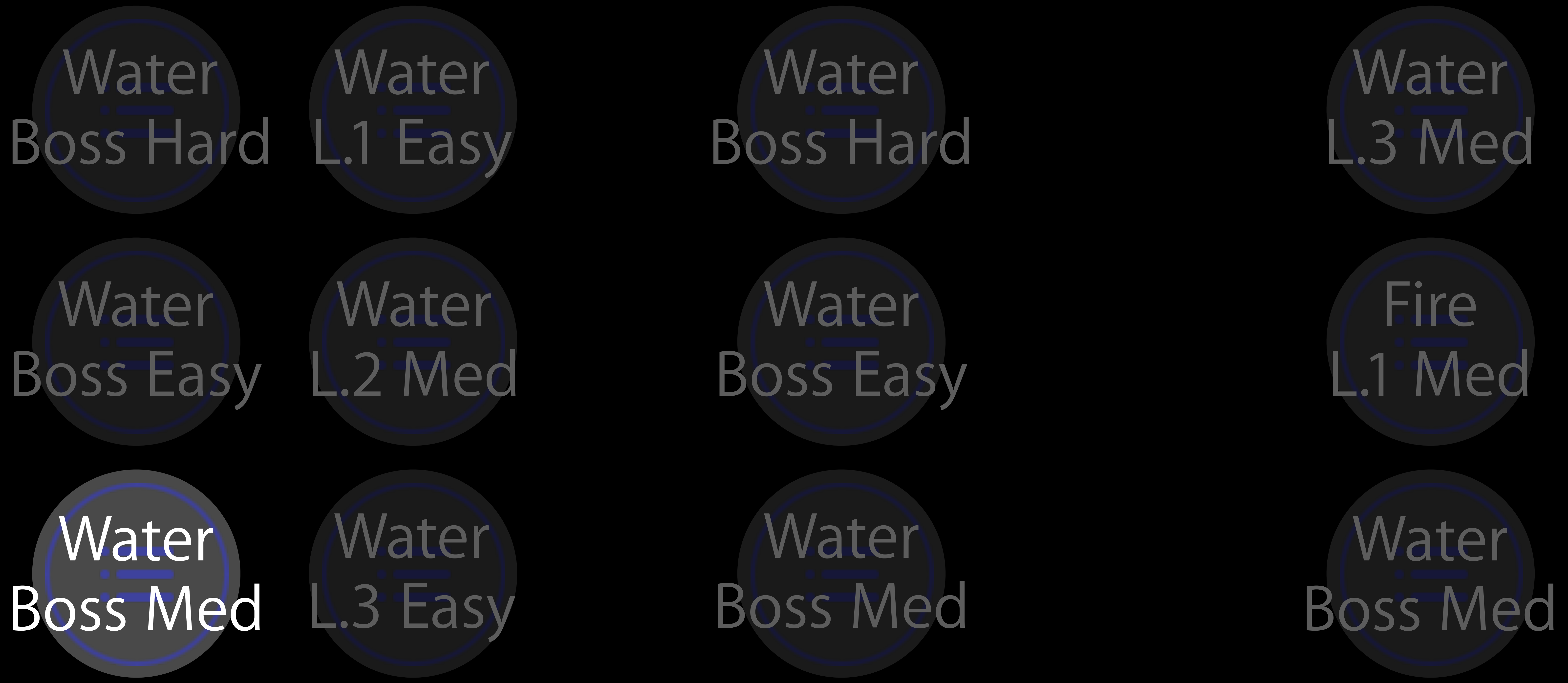
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Water Boss Hard

Water L.1 Easy

Water Boss Easy

Water L.2 Med

Water Boss Med

Water L.3 Easy

Water Planet

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss Set

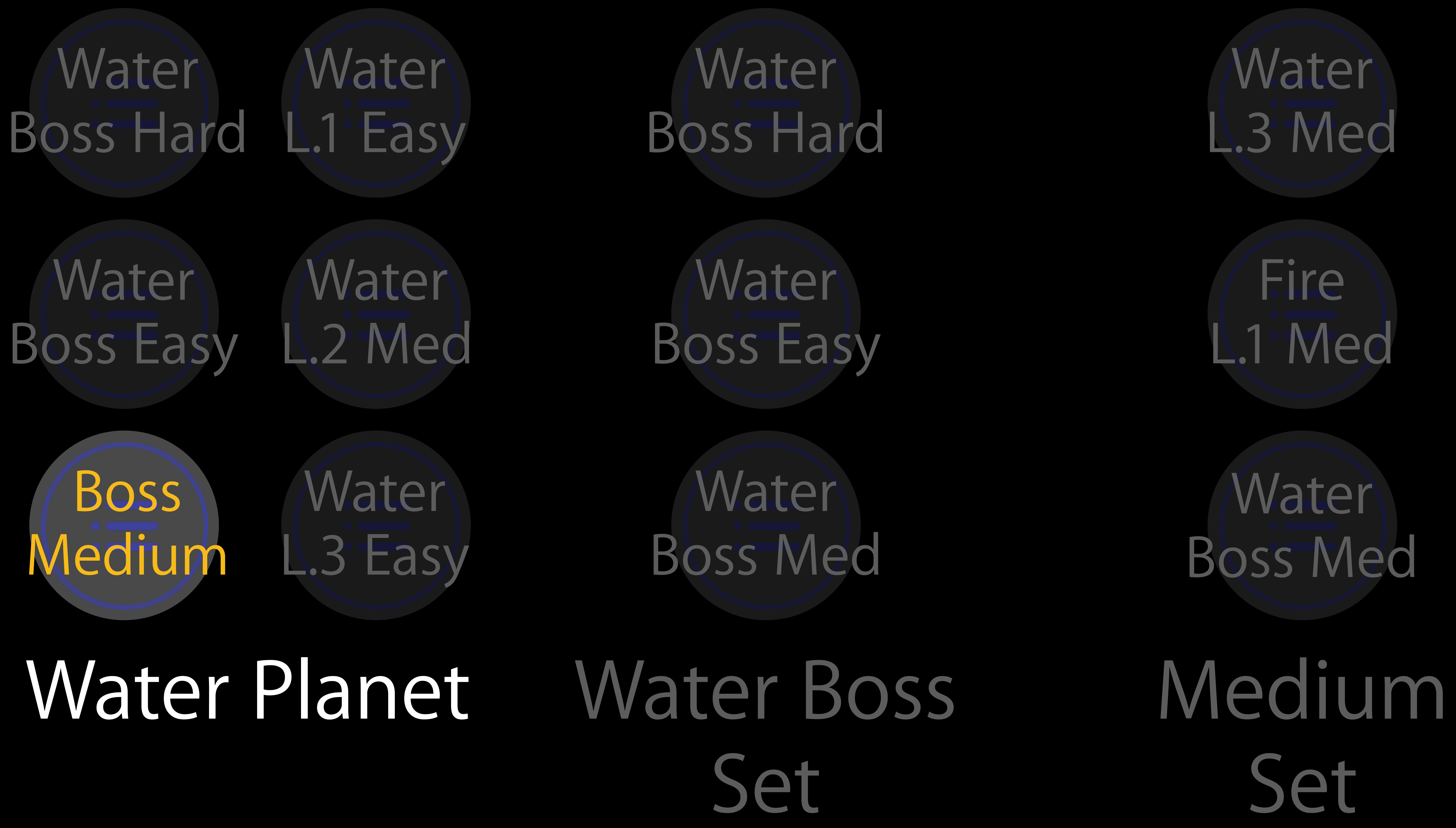
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Water Boss Hard

Water L.1 Easy

Water Boss Easy

Water L.2 Med

Boss Medium

Water L.3 Easy

Water Planet

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss Set

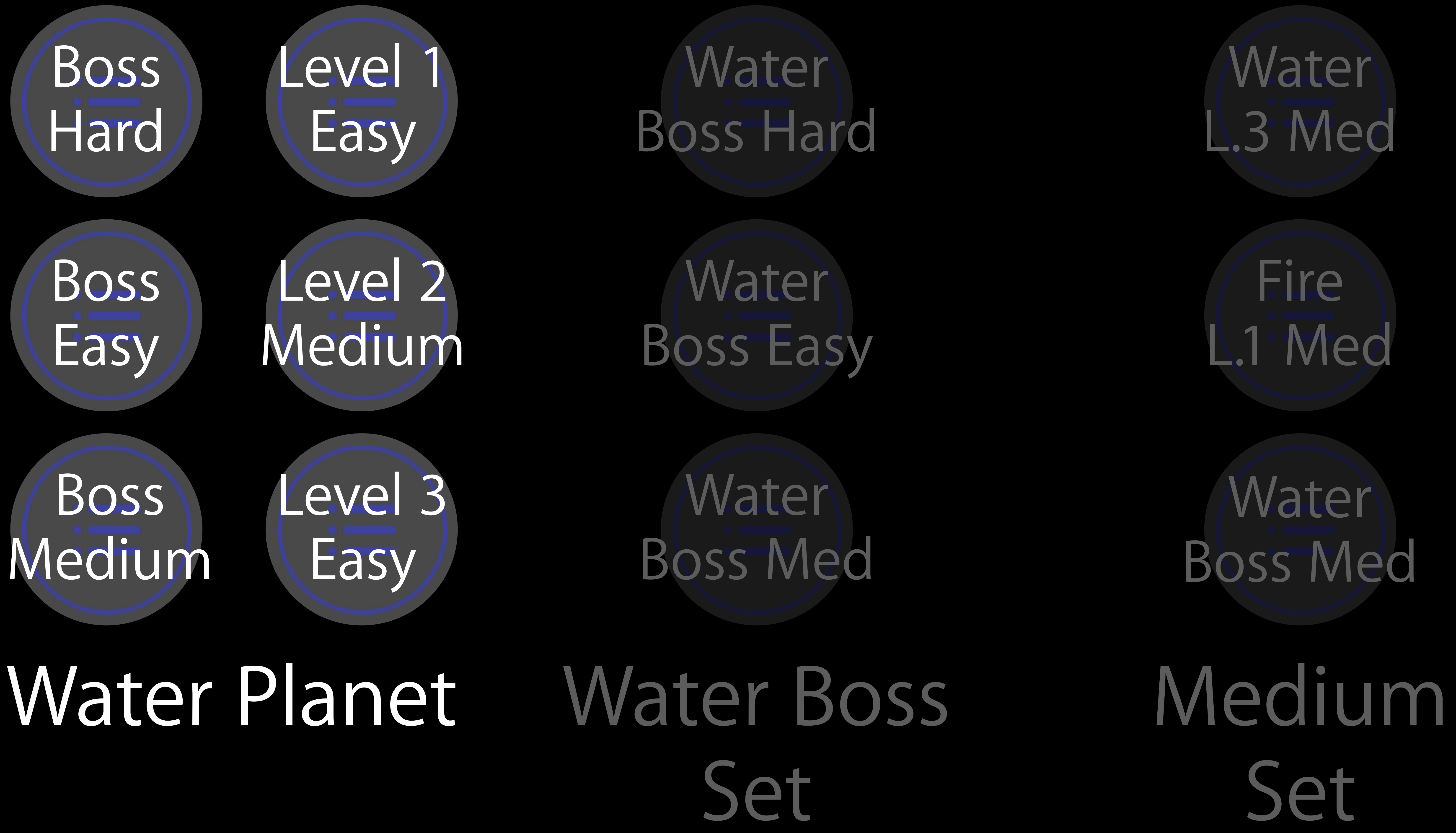
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Boss Hard

Level 1 Easy

Boss Easy

Level 2 Medium

Boss Medium

Level 3 Easy

Water Planet

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss Set

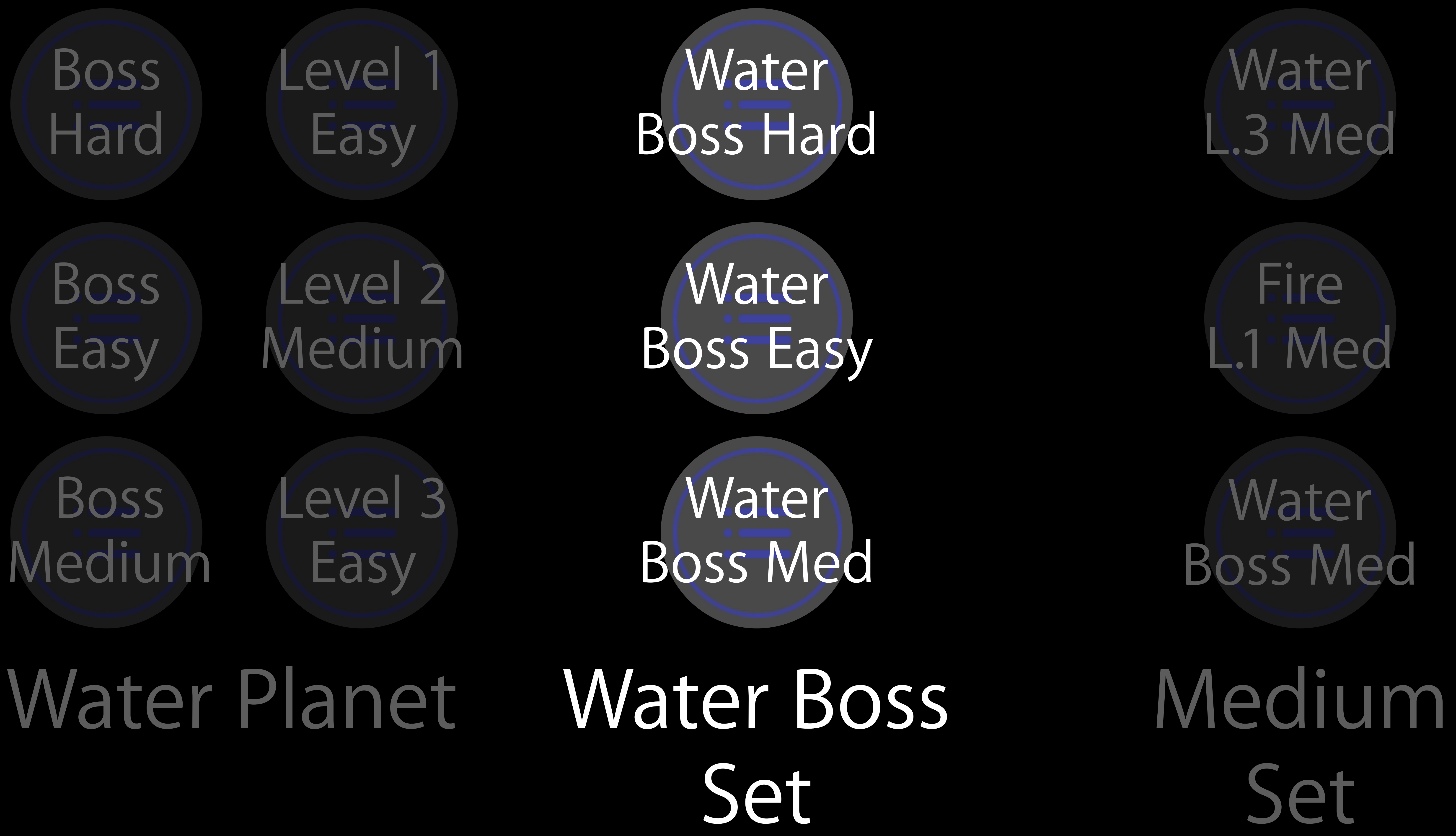
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Boss Hard
Boss Easy
Boss Medium
Level 1 Easy
Level 2 Medium
Level 3 Easy

Water Planet

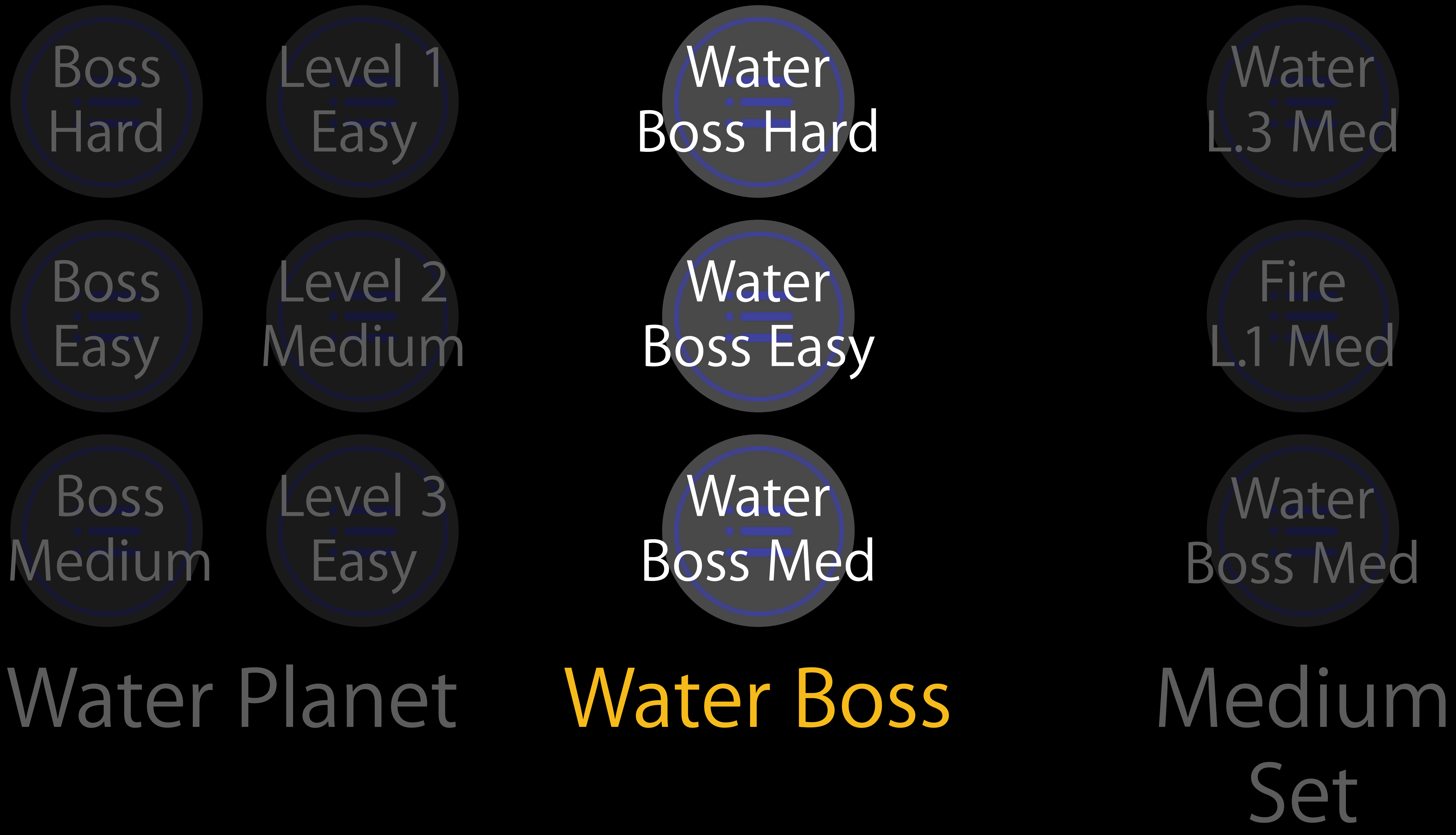
Water Boss Hard
Water Boss Easy
Water Boss Med

Water Boss Set

Water L.3 Med
Fire L.1 Med
Water Boss Med

Medium Set

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss

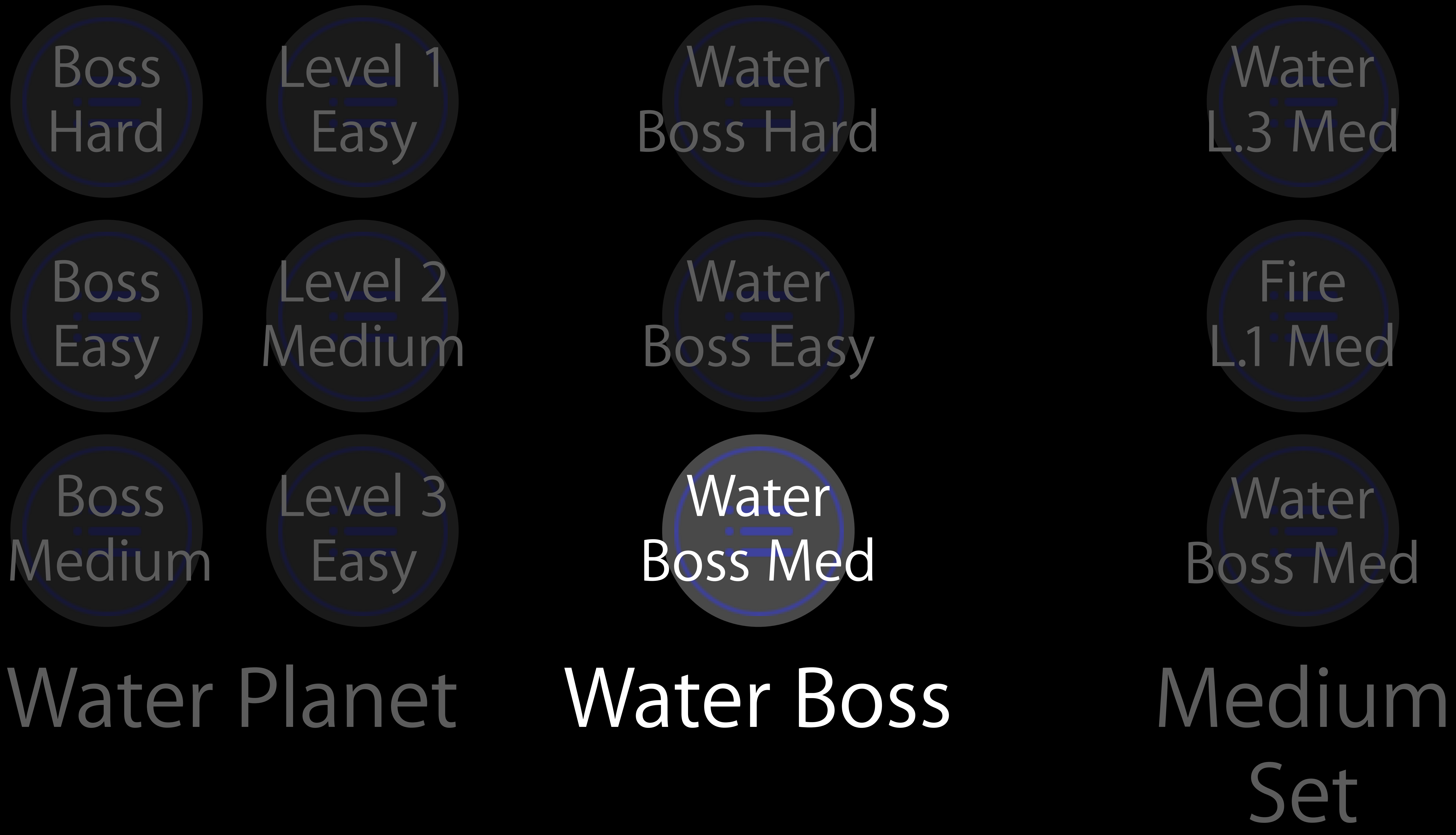
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Water Boss Hard

Water Boss Easy

Water Boss Med

Water Boss

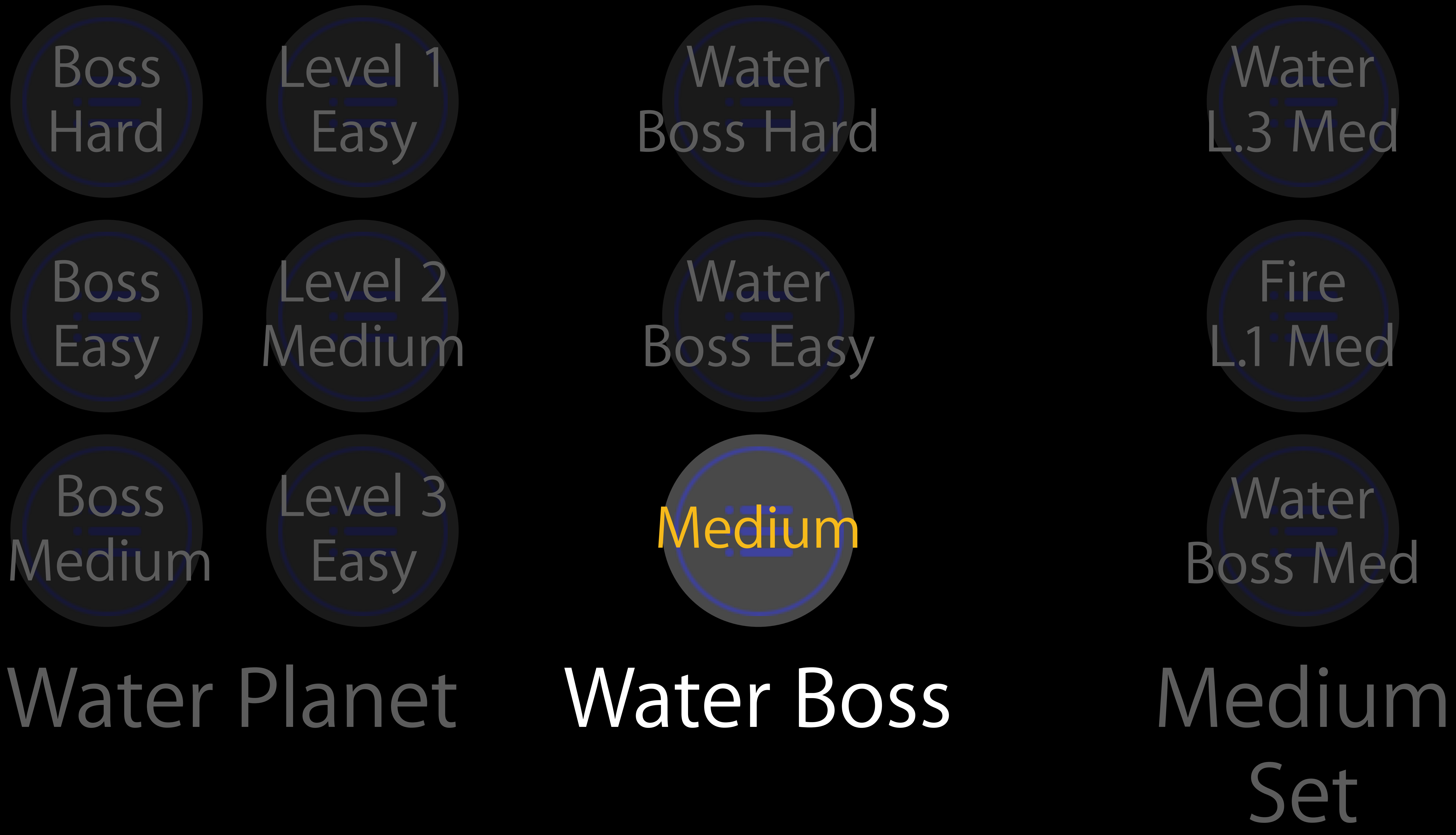
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Water Boss Hard

Water Boss Easy

Medium

Water Boss

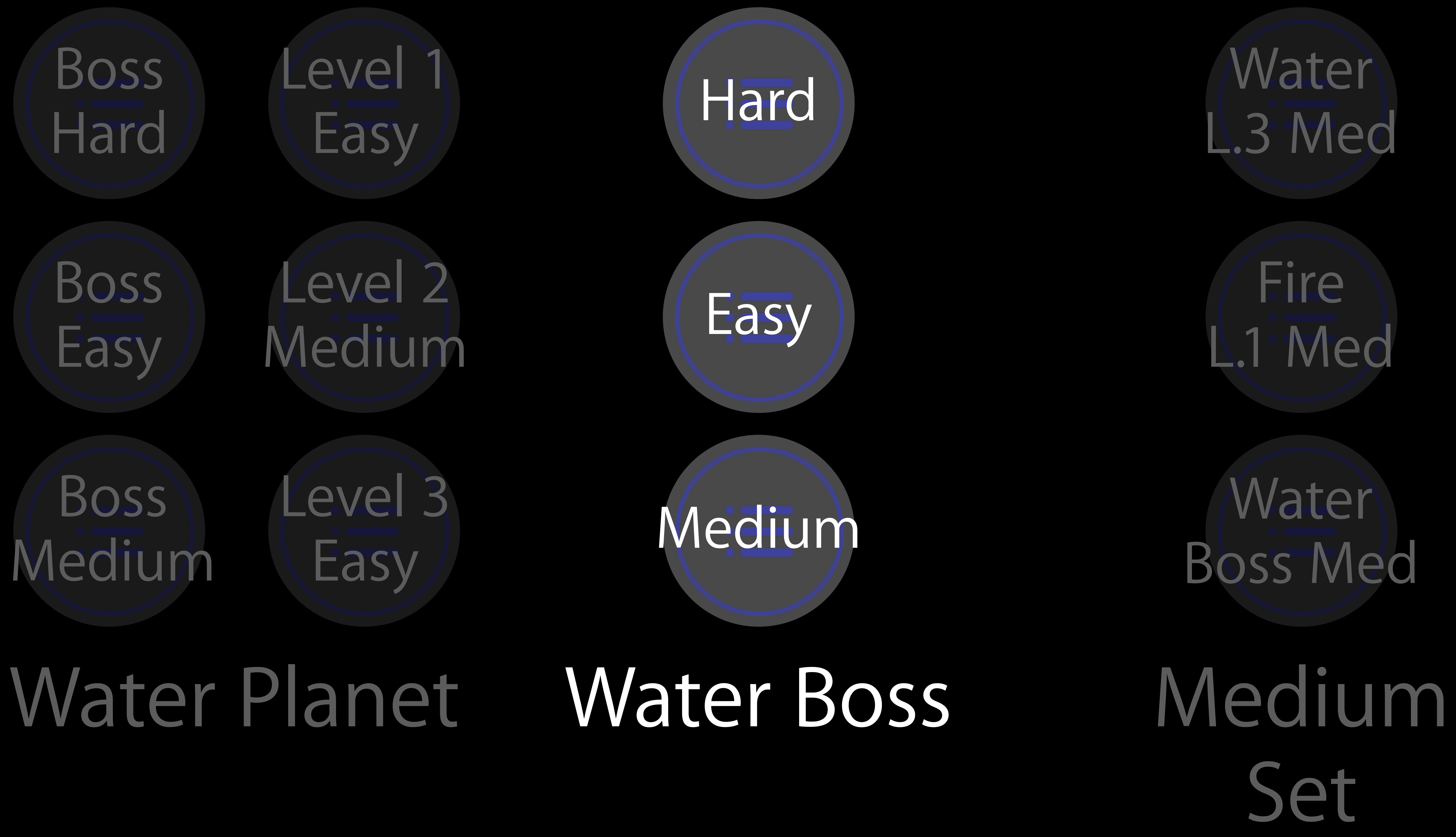
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Hard

Easy

Medium

Water Boss

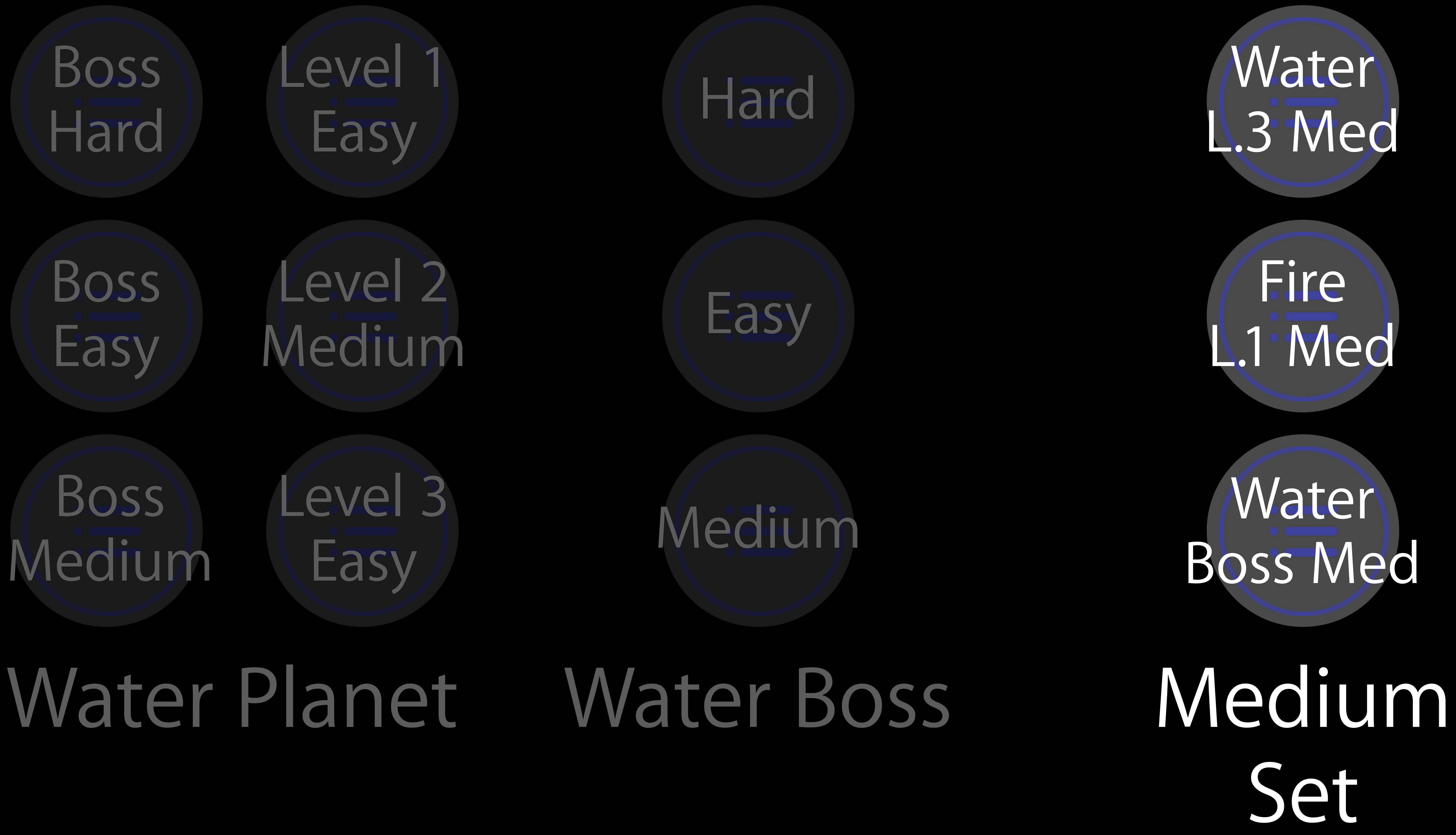
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Hard

Easy

Medium

Water Boss

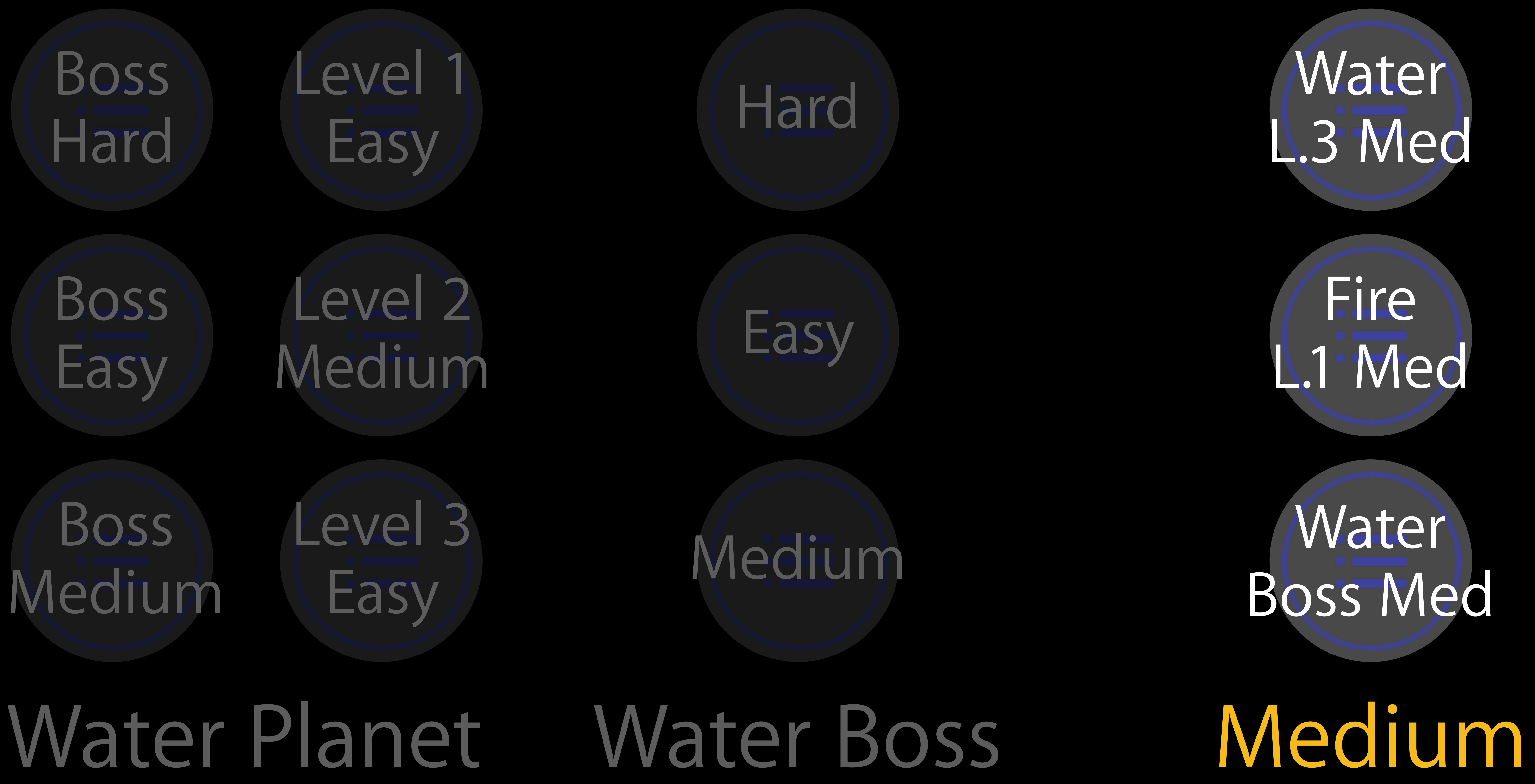
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium Set

Naming Sets



Boss Hard

Level 1 Easy

Boss Easy

Level 2 Medium

Boss Medium

Level 3 Easy

Water Planet

Hard

Easy

Medium

Water Boss

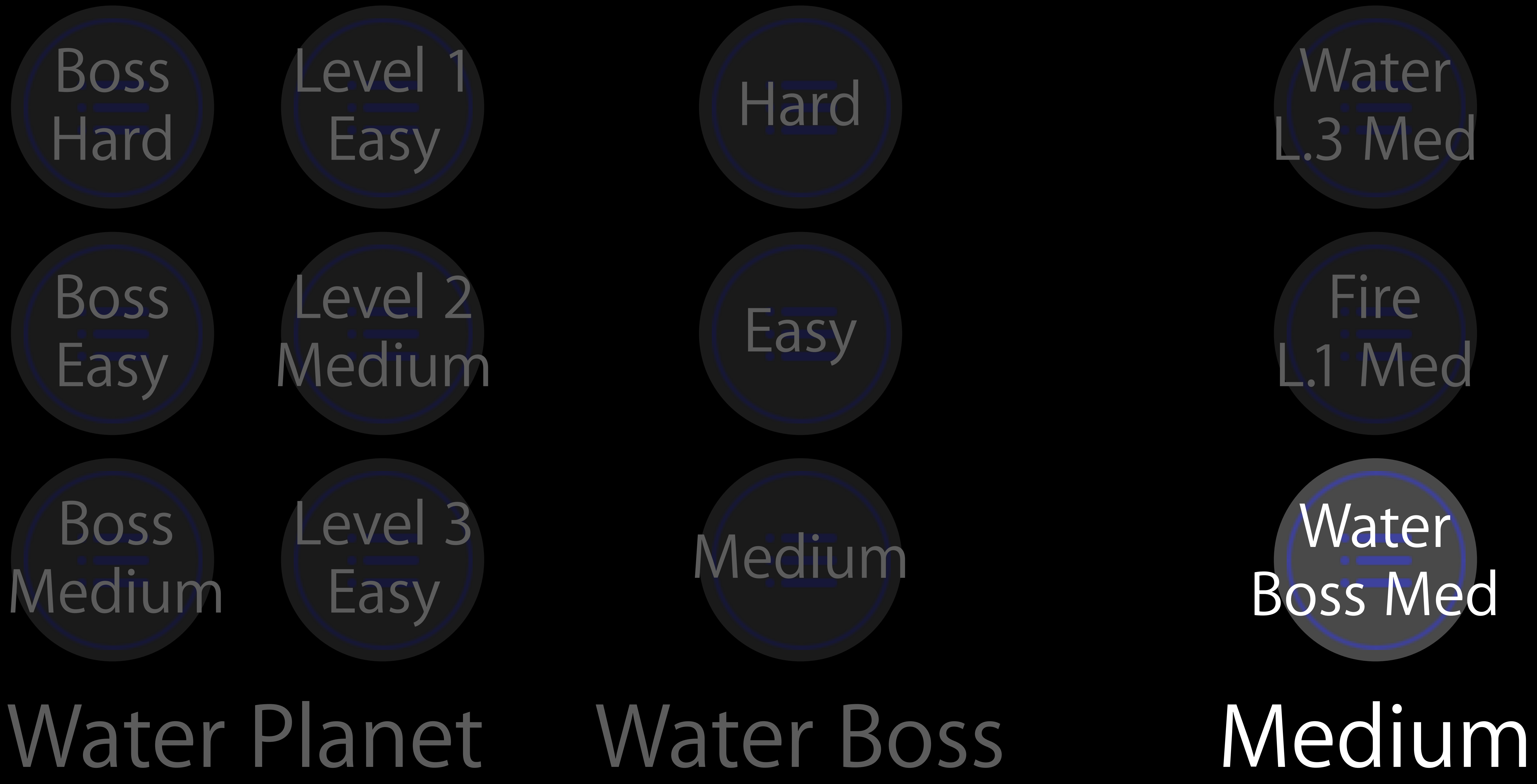
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Hard

Easy

Medium

Water Boss

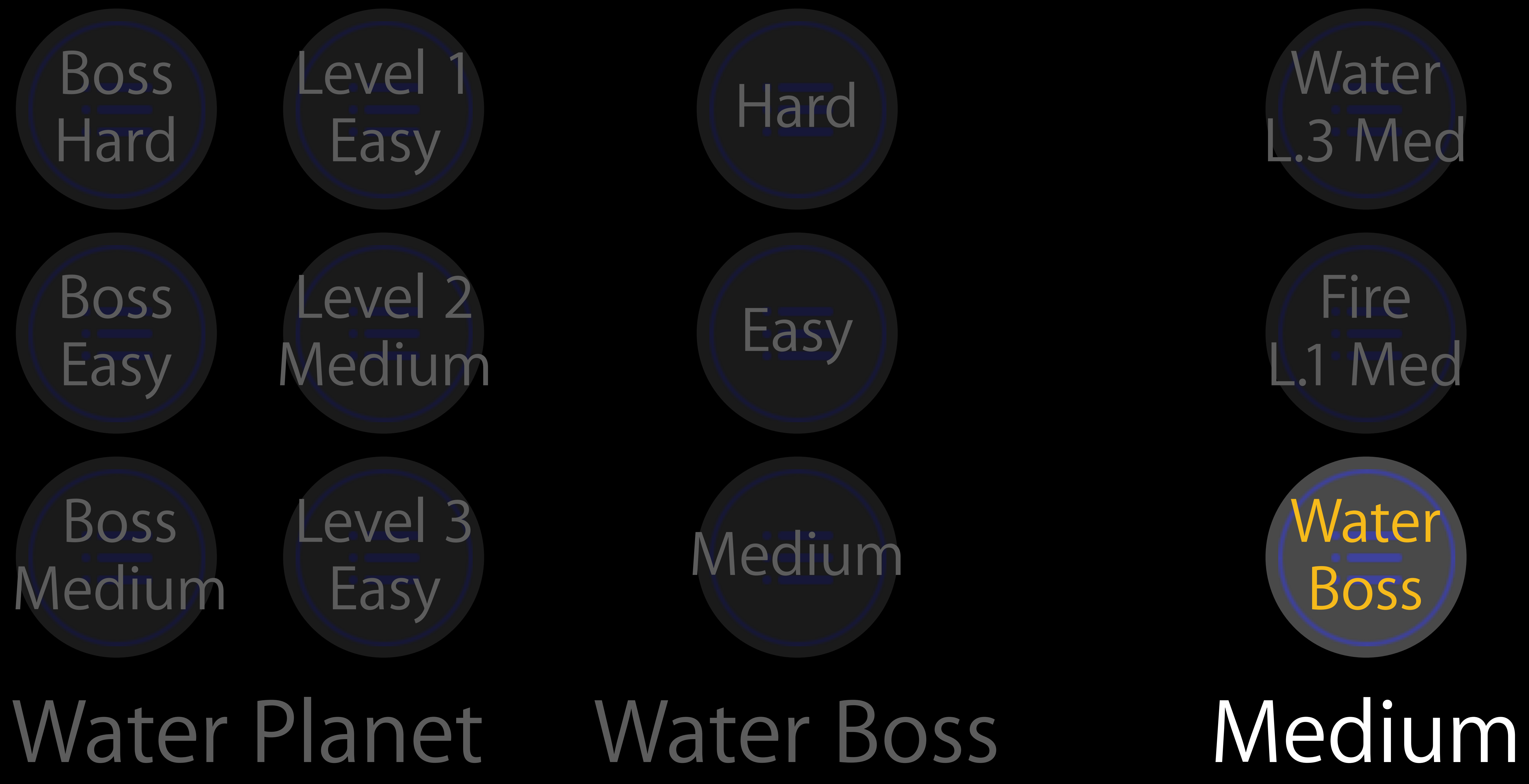
Water L.3 Med

Fire L.1 Med

Water Boss Med

Medium

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Hard

Easy

Medium

Water Boss

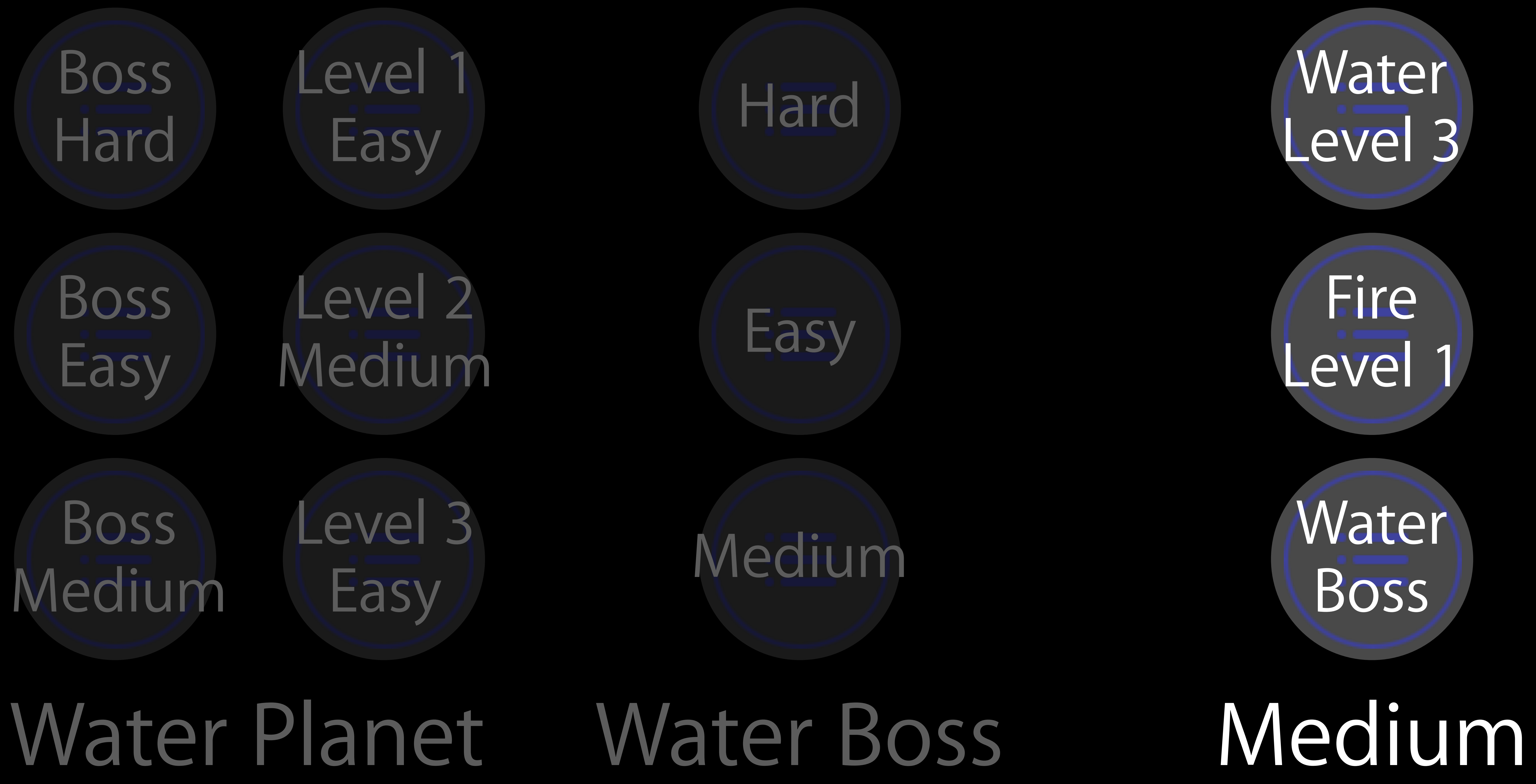
Water L.3 Med

Fire L.1 Med

Water Boss

Medium

Naming Sets



Boss Hard

Level 1 Easy

Boss Easy

Level 2 Medium

Boss Medium

Level 3 Easy

Water Planet

Hard

Easy

Medium

Water Boss

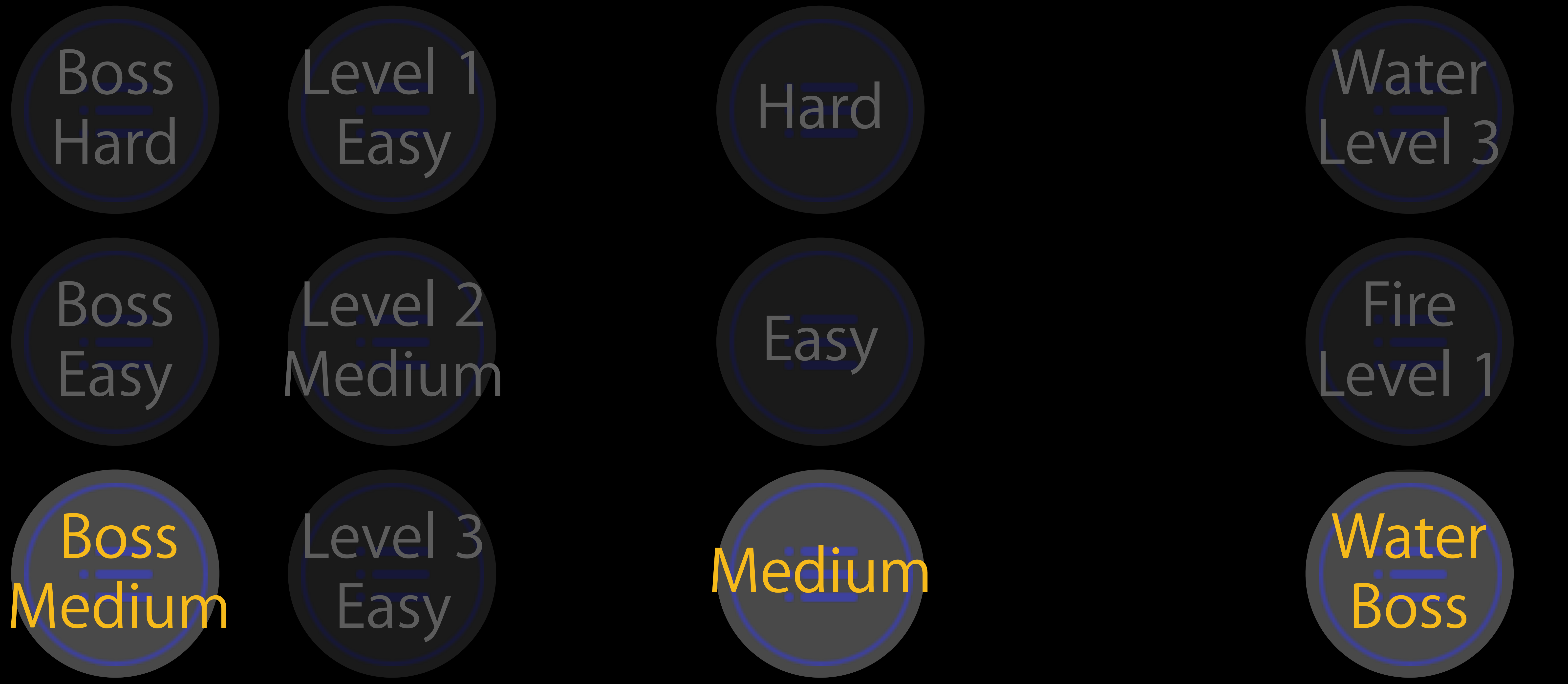
Water Level 3

Fire Level 1

Water Boss

Medium

Naming Sets



Boss Hard

Boss Easy

Boss Medium

Level 1 Easy

Level 2 Medium

Level 3 Easy

Water Planet

Hard

Easy

Medium

Water Boss

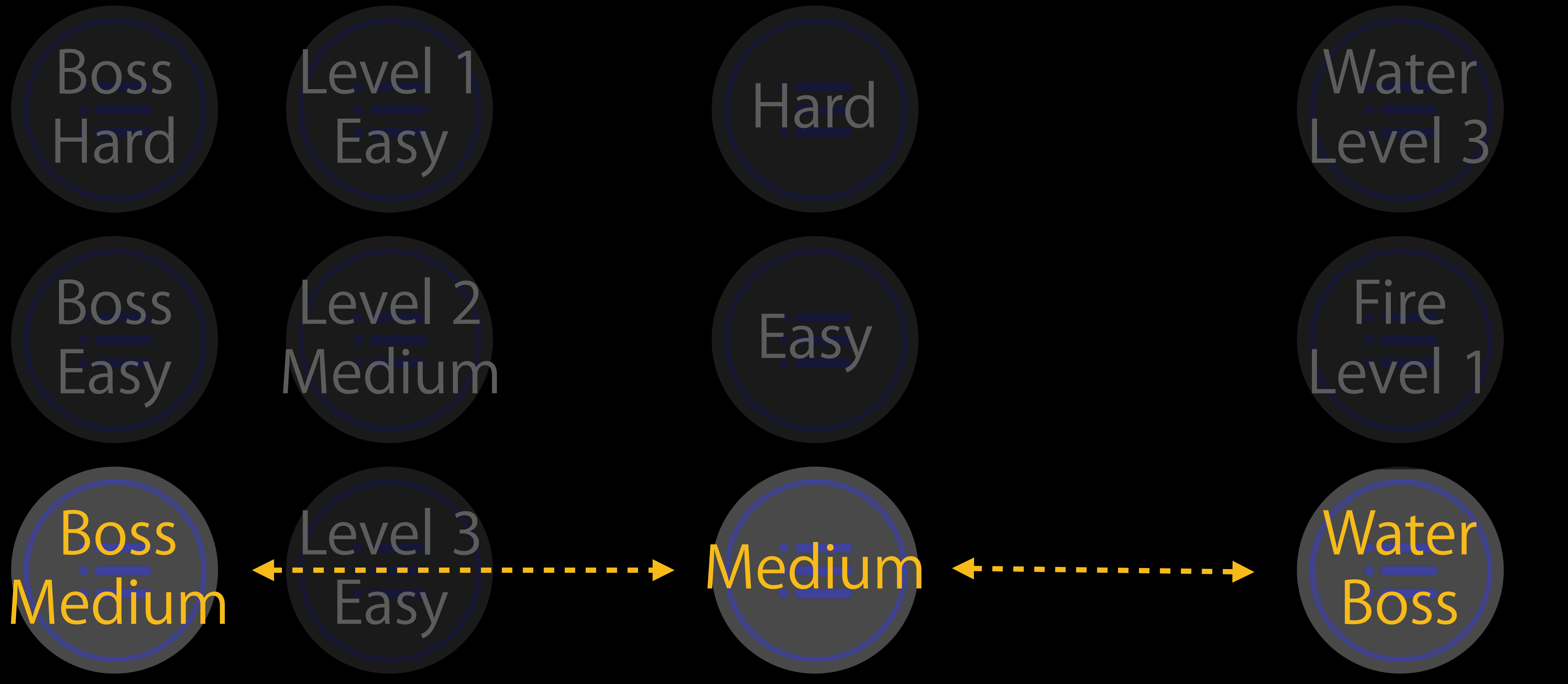
Water Level 3

Fire Level 1

Water Boss

Medium

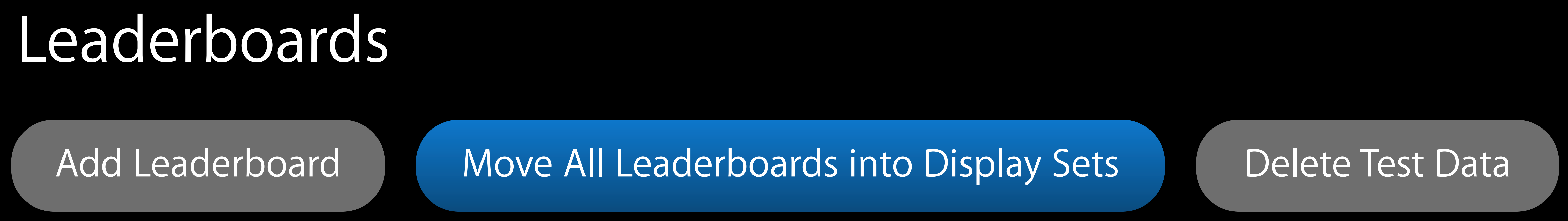
Naming Sets



Water Planet   Water Boss   Medium

iTunes Connect Setup

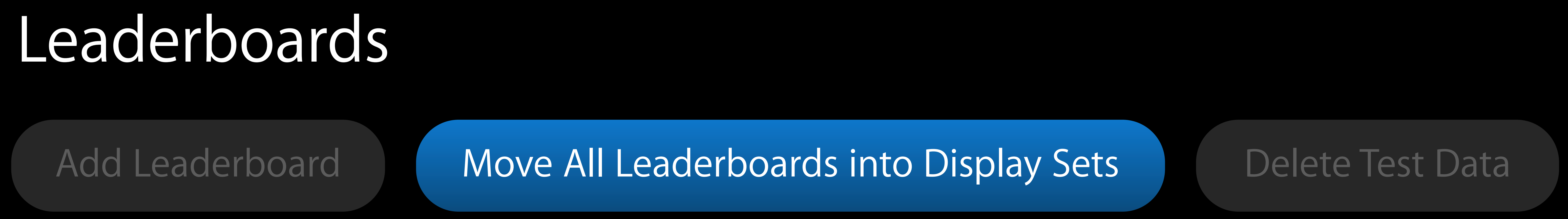
Quick flyover



Leaderboards

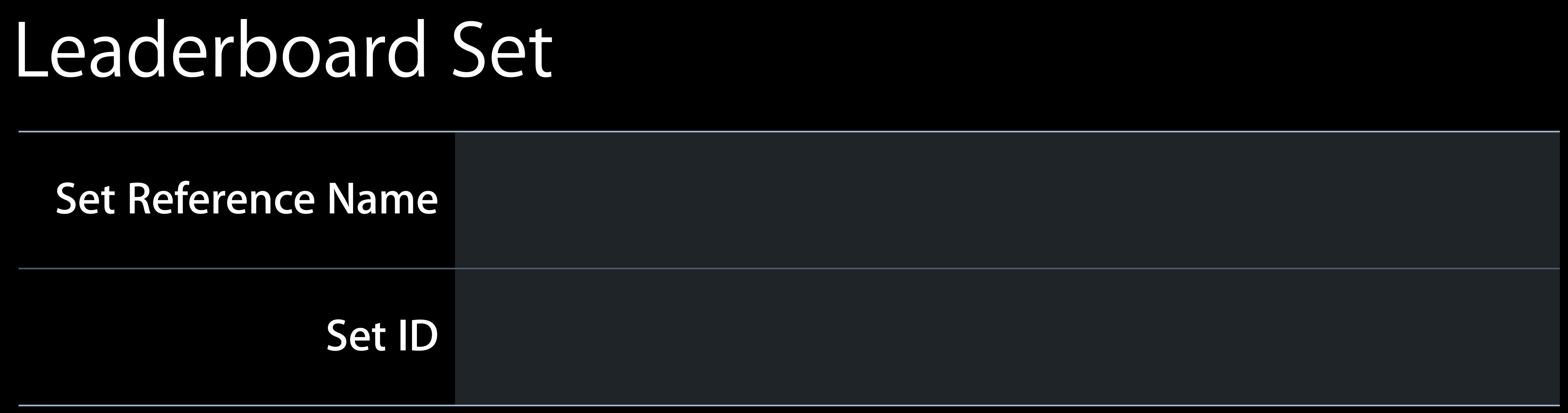
iTunes Connect Setup

Quick flyover



iTunes Connect Setup

Quick flyover

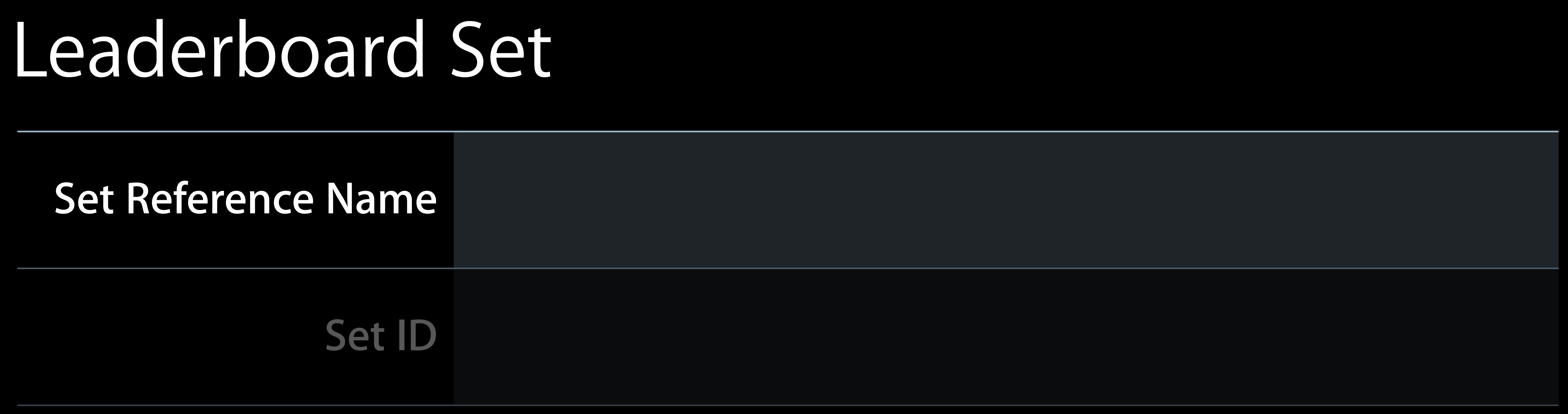


Leaderboard Set

Set Reference Name	
Set ID	

iTunes Connect Setup

Quick flyover

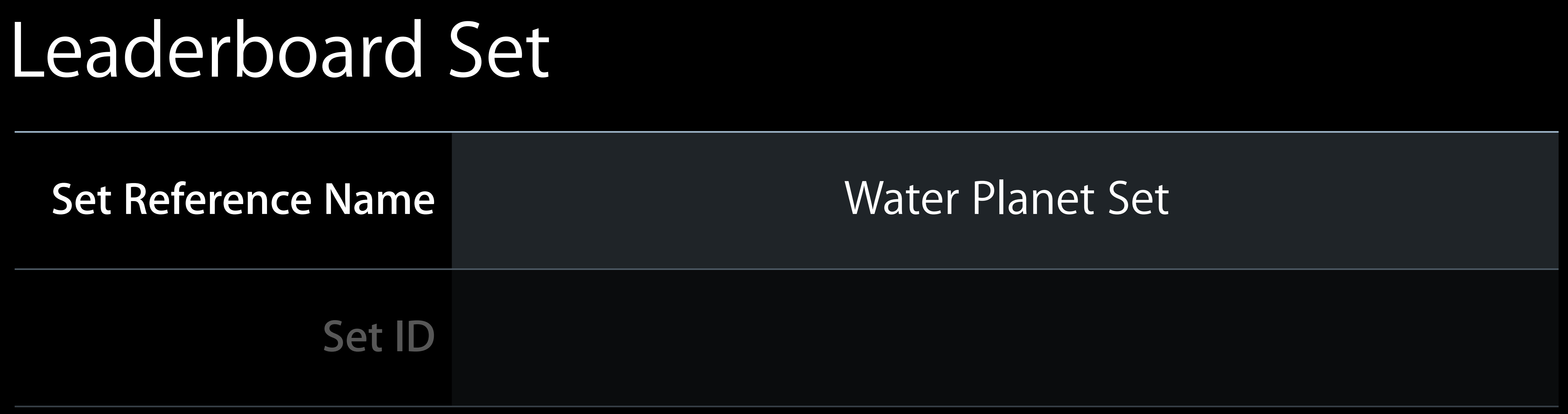


Leaderboard Set

Set Reference Name	
Set ID	

iTunes Connect Setup

Quick flyover

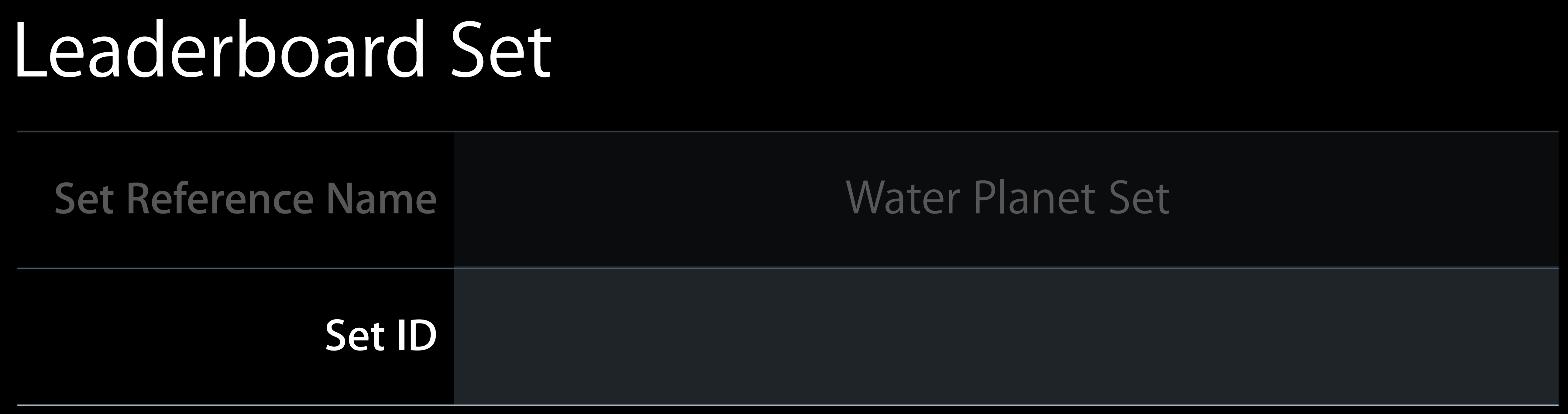


Leaderboard Set

Set Reference Name	Water Planet Set
Set ID	

iTunes Connect Setup

Quick flyover

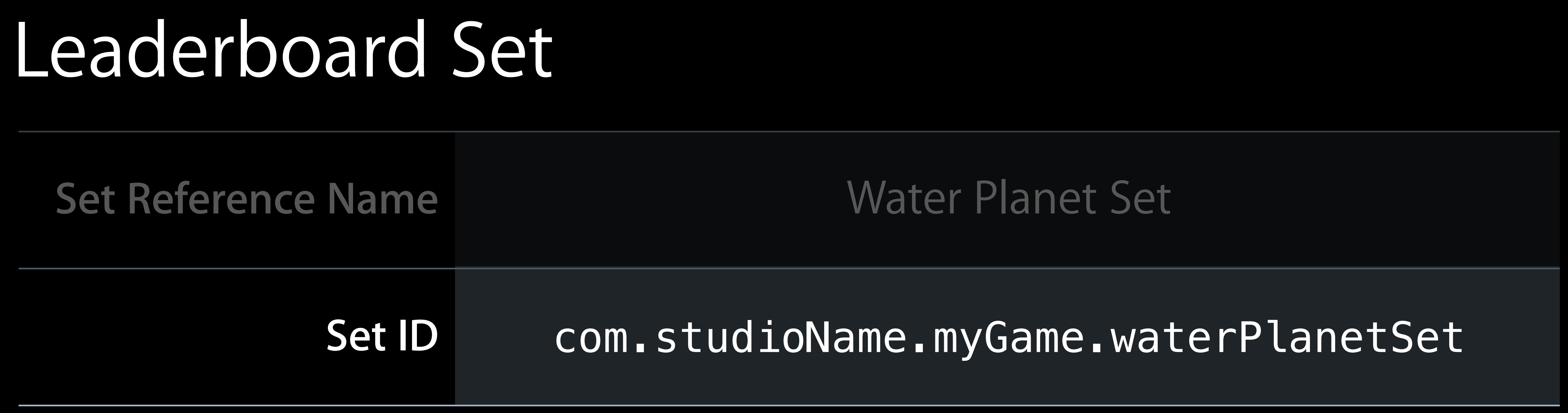


Leaderboard Set

Set Reference Name	Water Planet Set
Set ID	

iTunes Connect Setup

Quick flyover

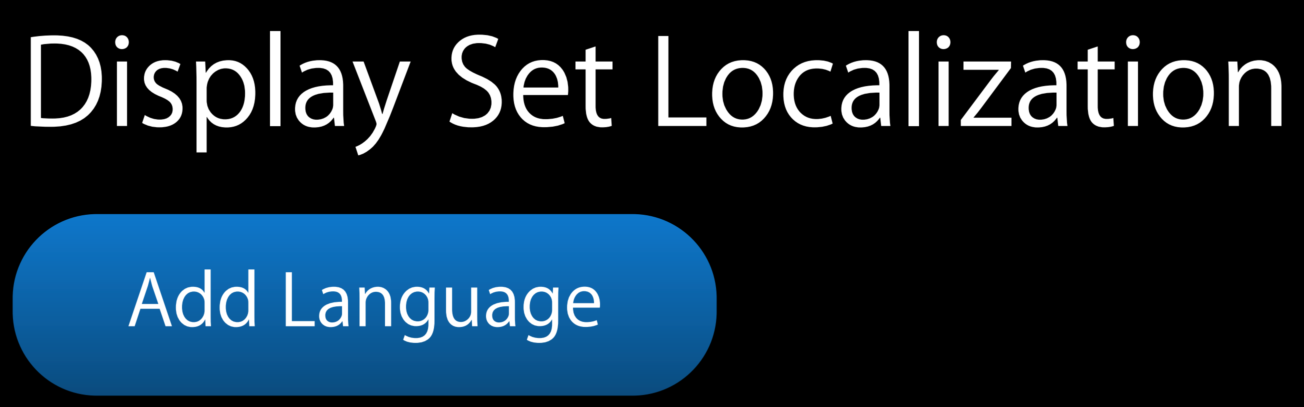


Leaderboard Set

Set Reference Name	Water Planet Set
Set ID	`com.studioName.myGame.waterPlanetSet`

iTunes Connect Setup

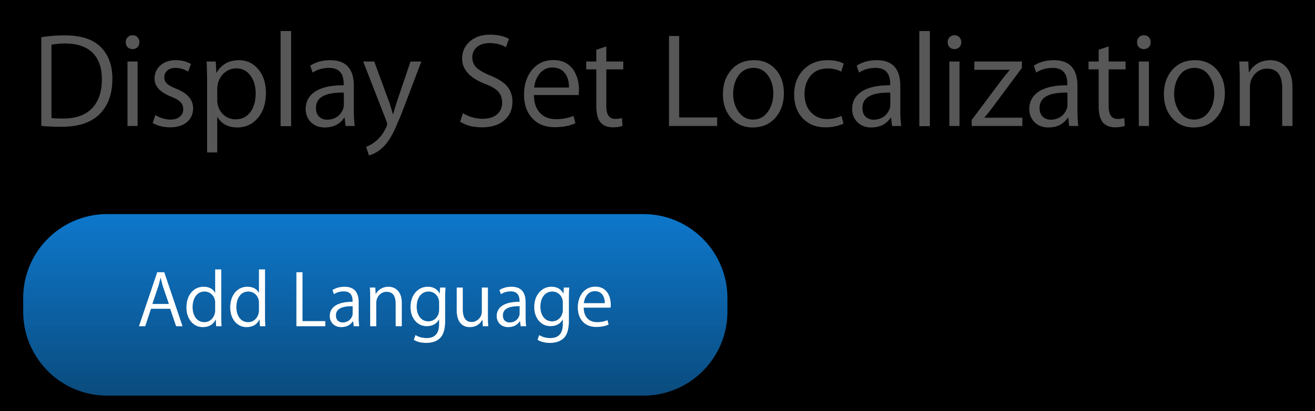
Quick flyover



Display Set Localization

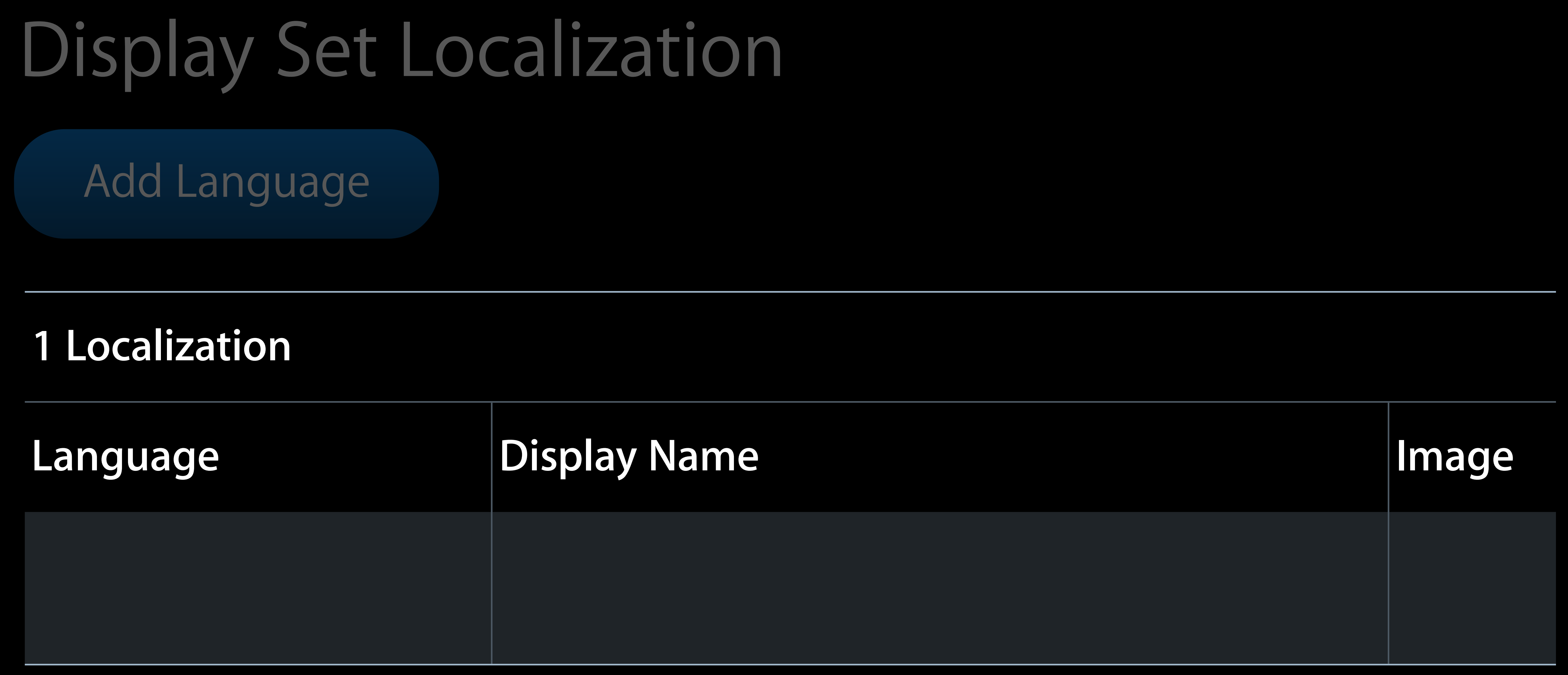
iTunes Connect Setup

Quick flyover



iTunes Connect Setup

Quick flyover



Display Set Localization

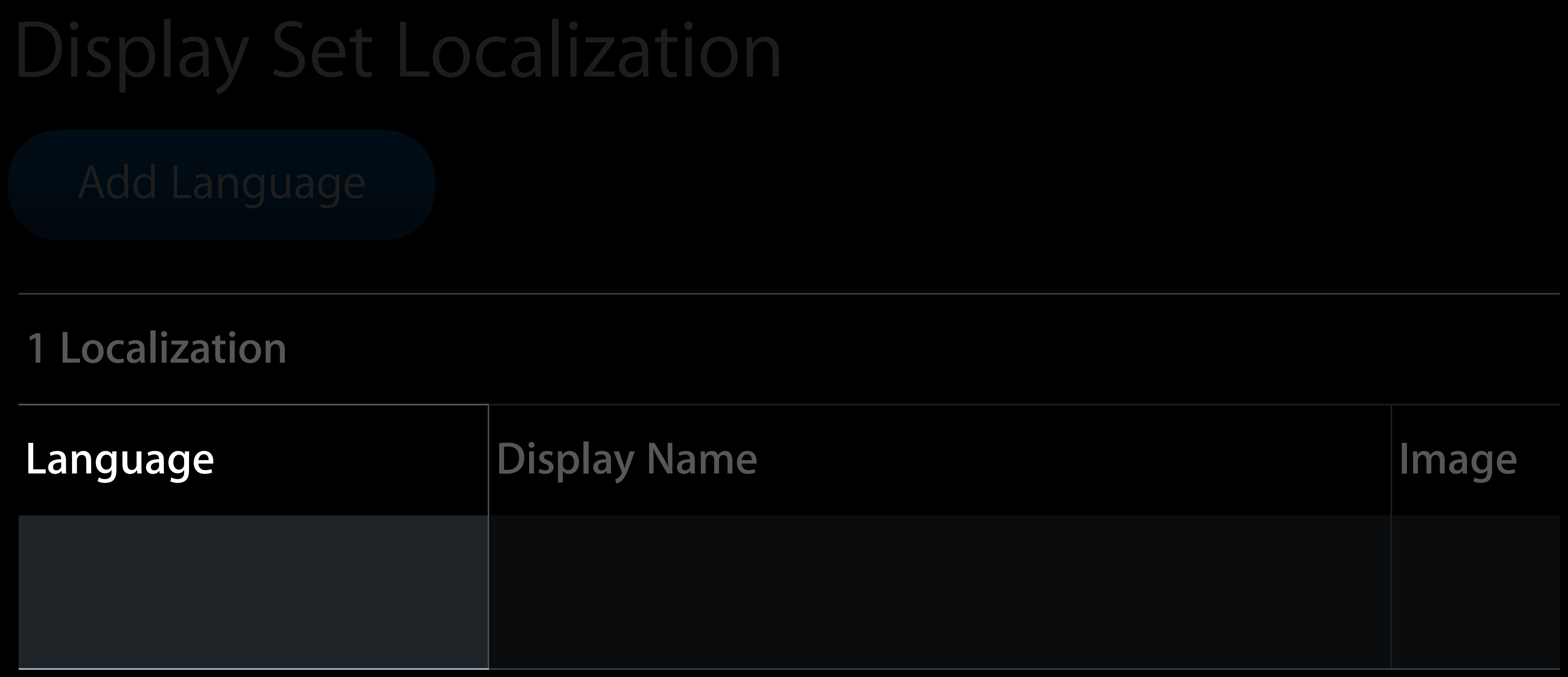
Add Language

1 Localization

Language	Display Name	Image

iTunes Connect Setup

Quick flyover



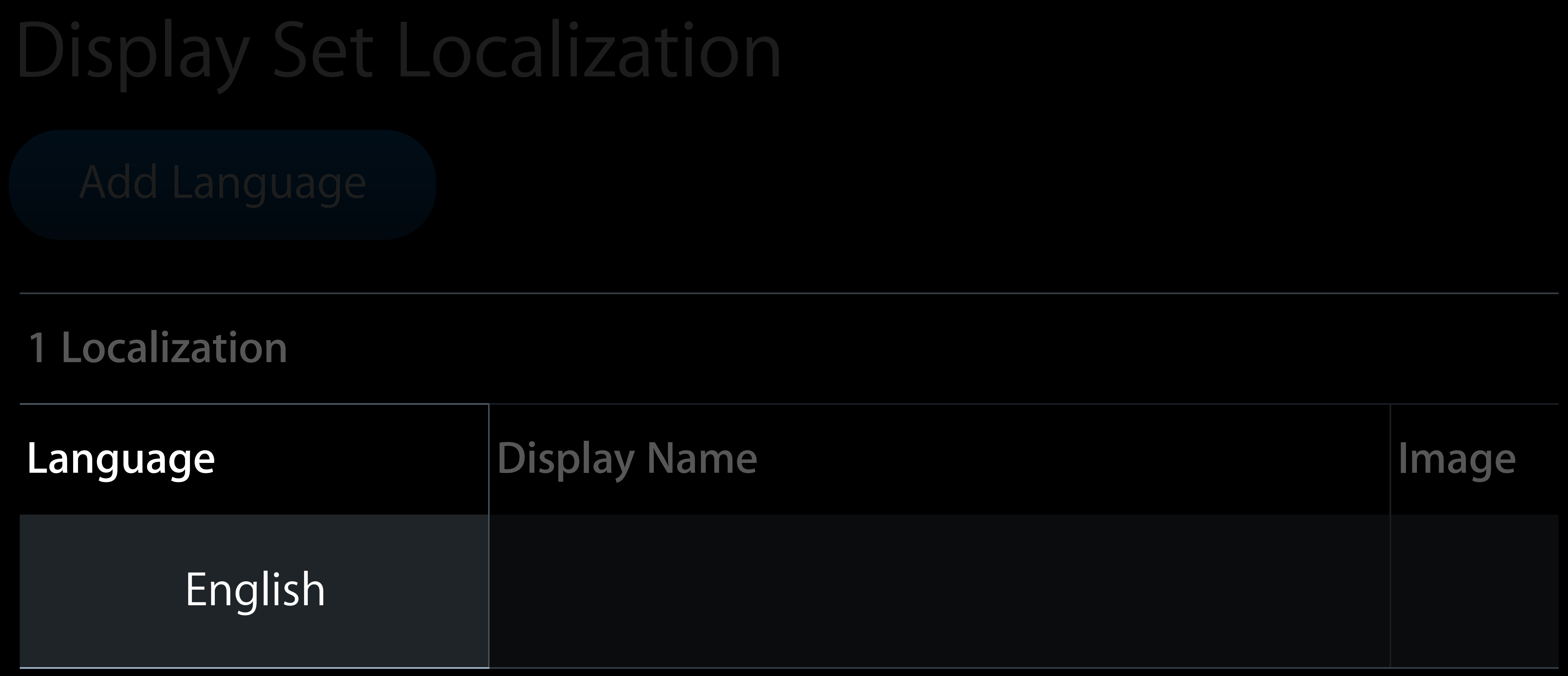
Display Set Localization

1 Localization

Language	Display Name	Image

iTunes Connect Setup

Quick flyover



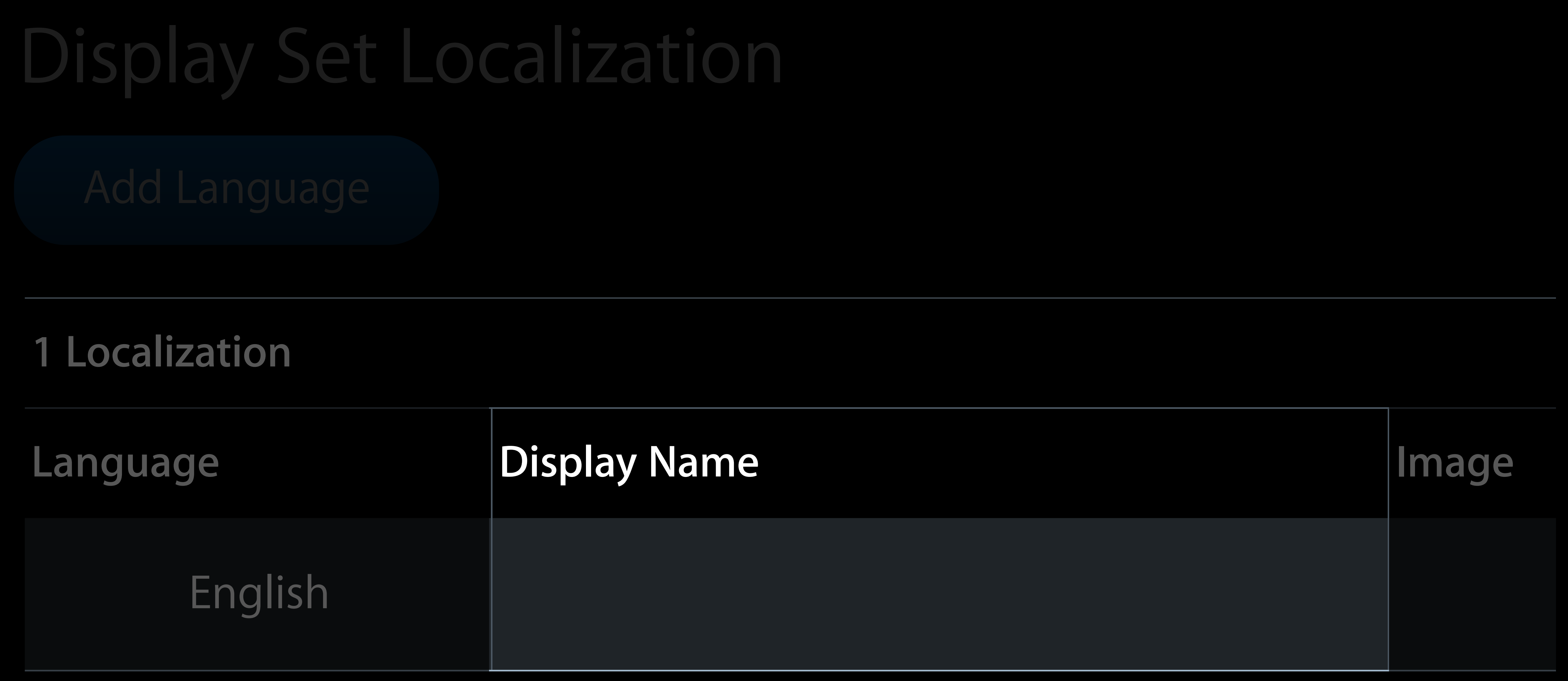
Display Set Localization

1 Localization

Language	Display Name	Image
English		

iTunes Connect Setup

Quick flyover



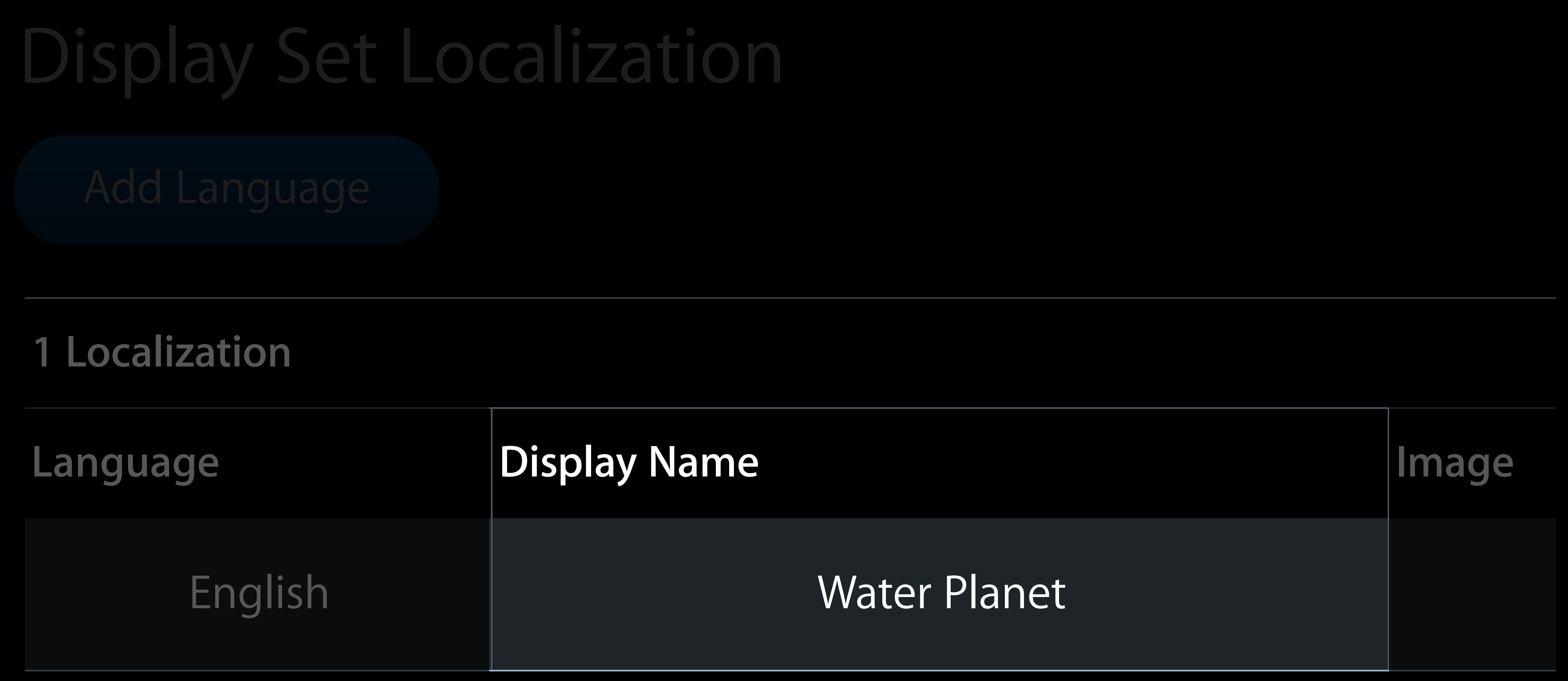
Display Set Localization

1 Localization

Language	Display Name	Image
English		

iTunes Connect Setup

Quick flyover



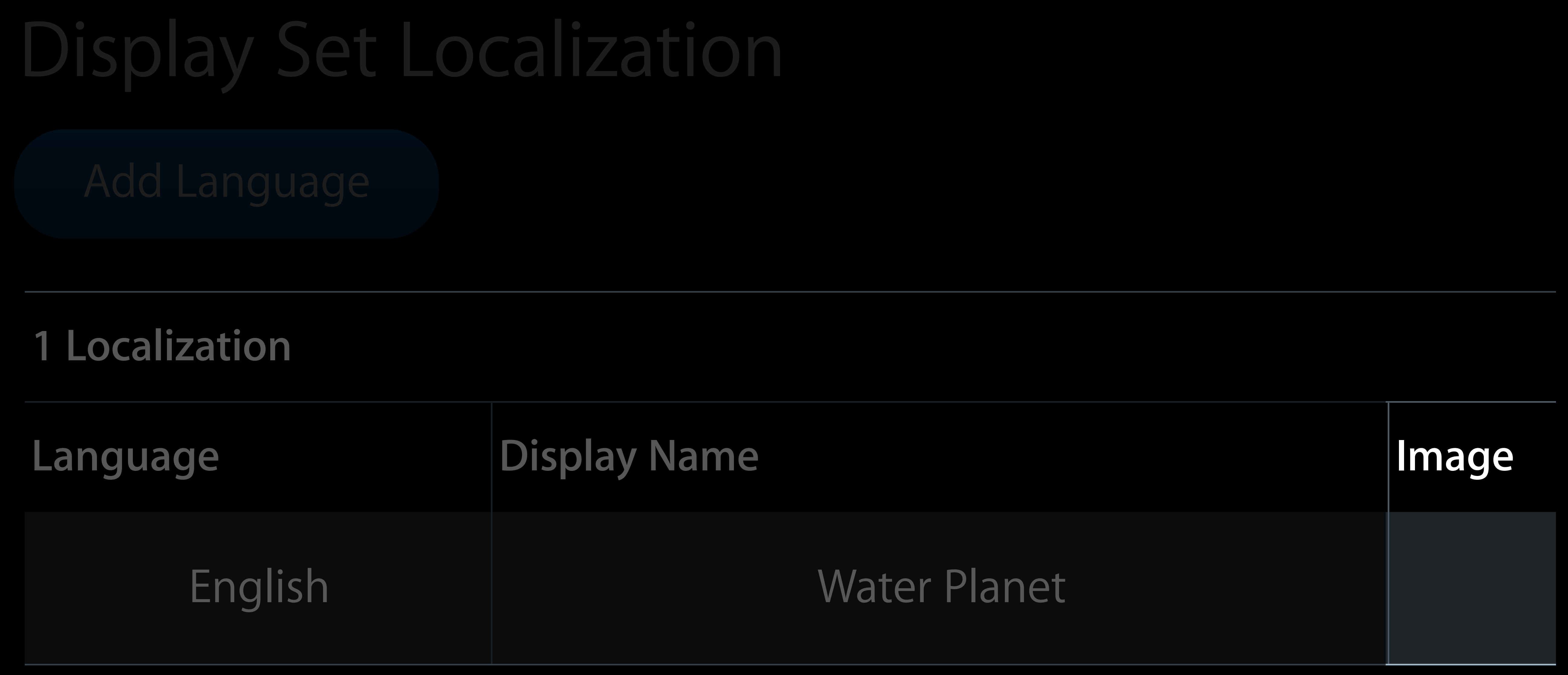
Display Set Localization

1 Localization

Language	Display Name	Image
English	Water Planet	

iTunes Connect Setup

Quick flyover



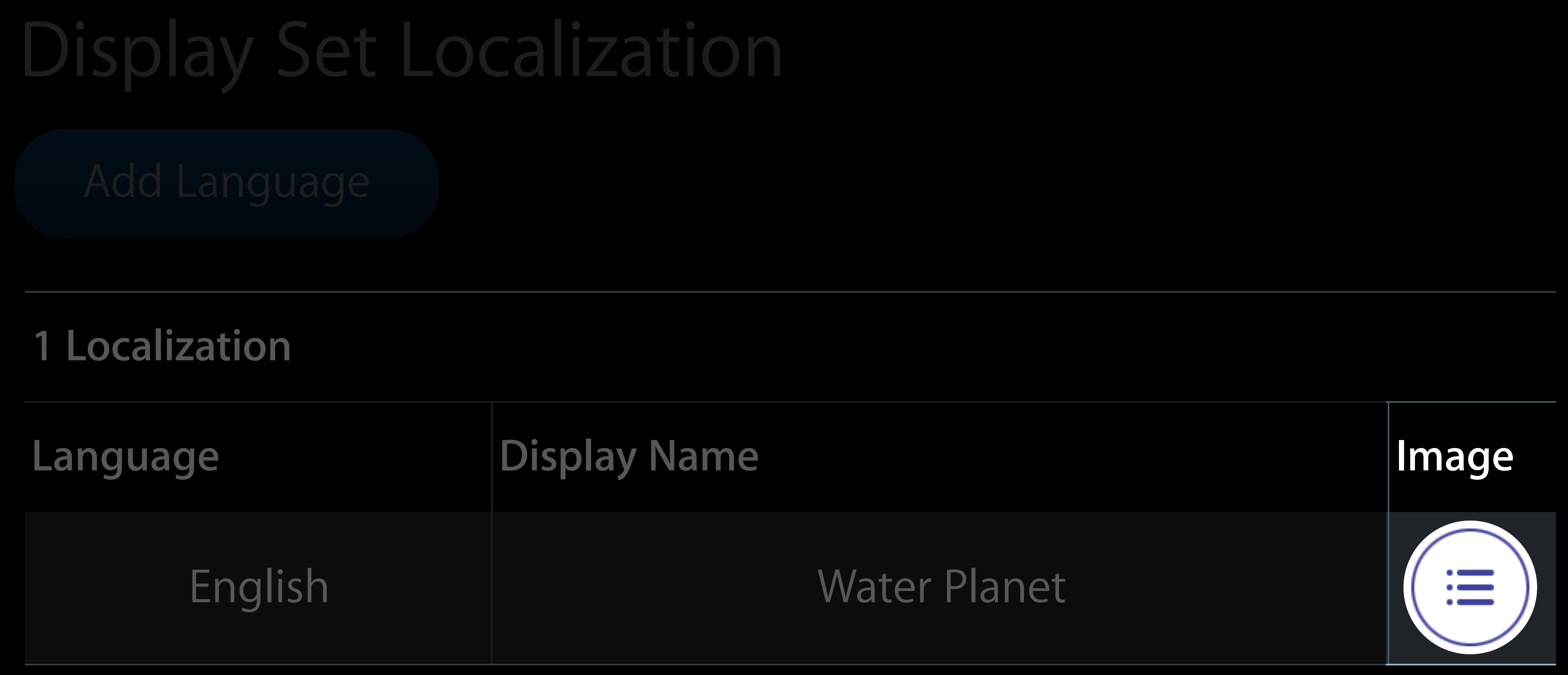
Display Set Localization

1 Localization

Language	Display Name	Image
English	Water Planet	

iTunes Connect Setup

Quick flyover



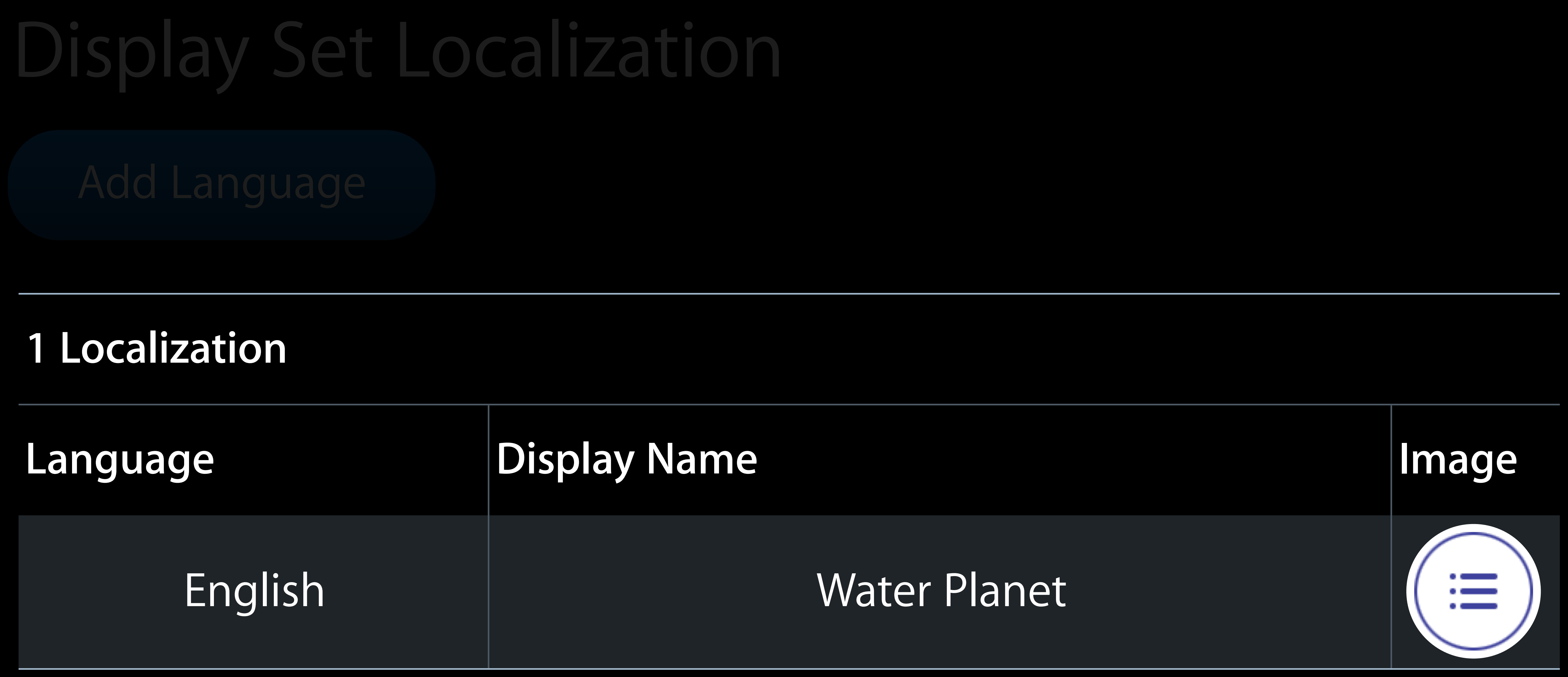
Display Set Localization

1 Localization

Language	Display Name	Image
English	Water Planet	☰

iTunes Connect Setup

Quick flyover



Display Set Localization

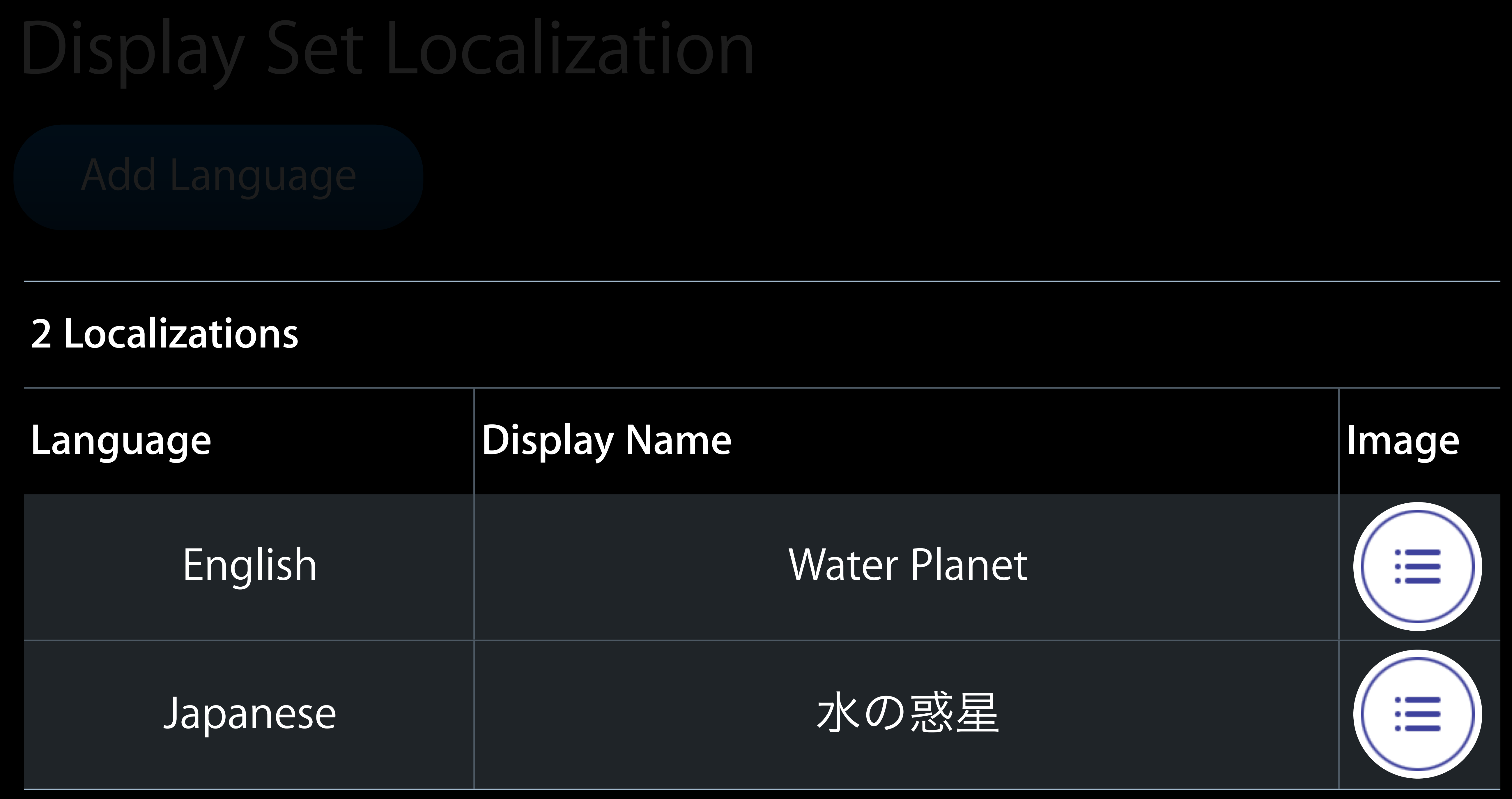
Add Language

1 Localization

Language	Display Name	Image
English	Water Planet	☰

iTunes Connect Setup

Quick flyover



Display Set Localization

2 Localizations

Language	Display Name	Image
English	Water Planet	☰
Japanese	水の惑星	☰

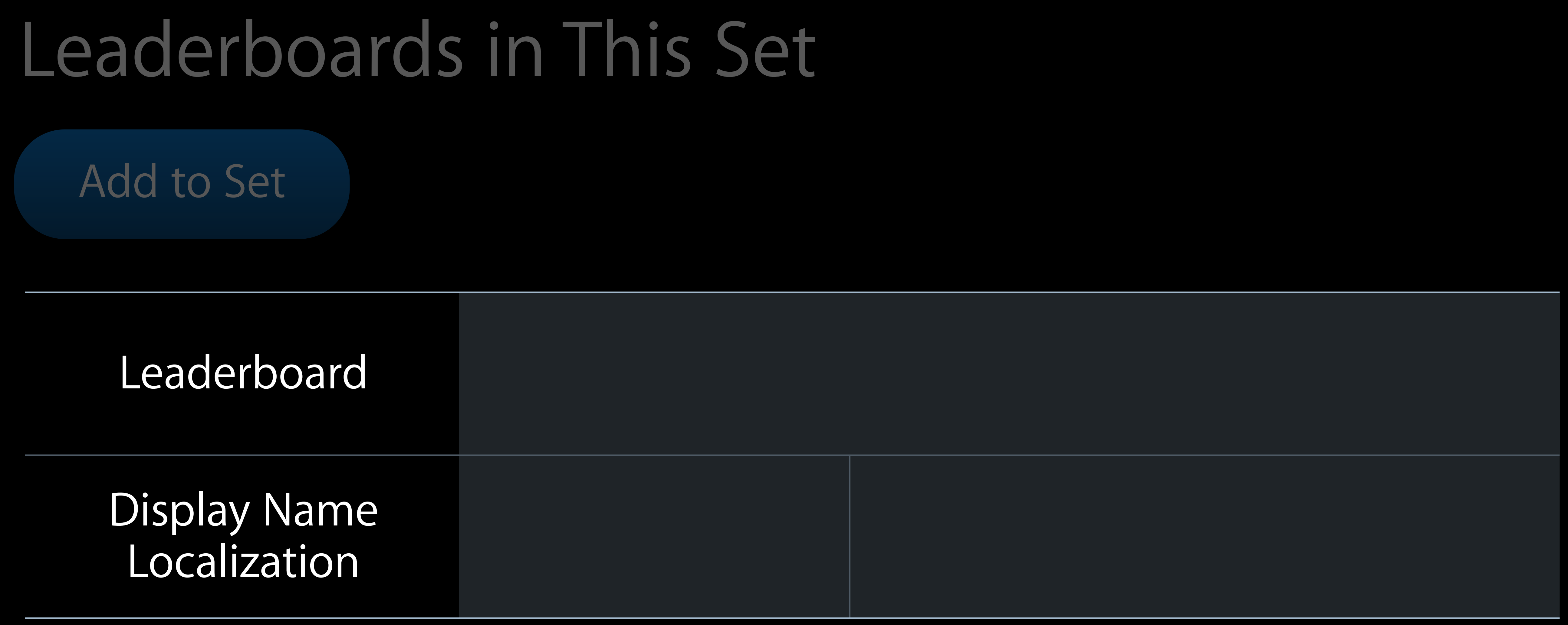
iTunes Connect Setup

Quick flyover

Leaderboards in This Set

iTunes Connect Setup

Quick flyover



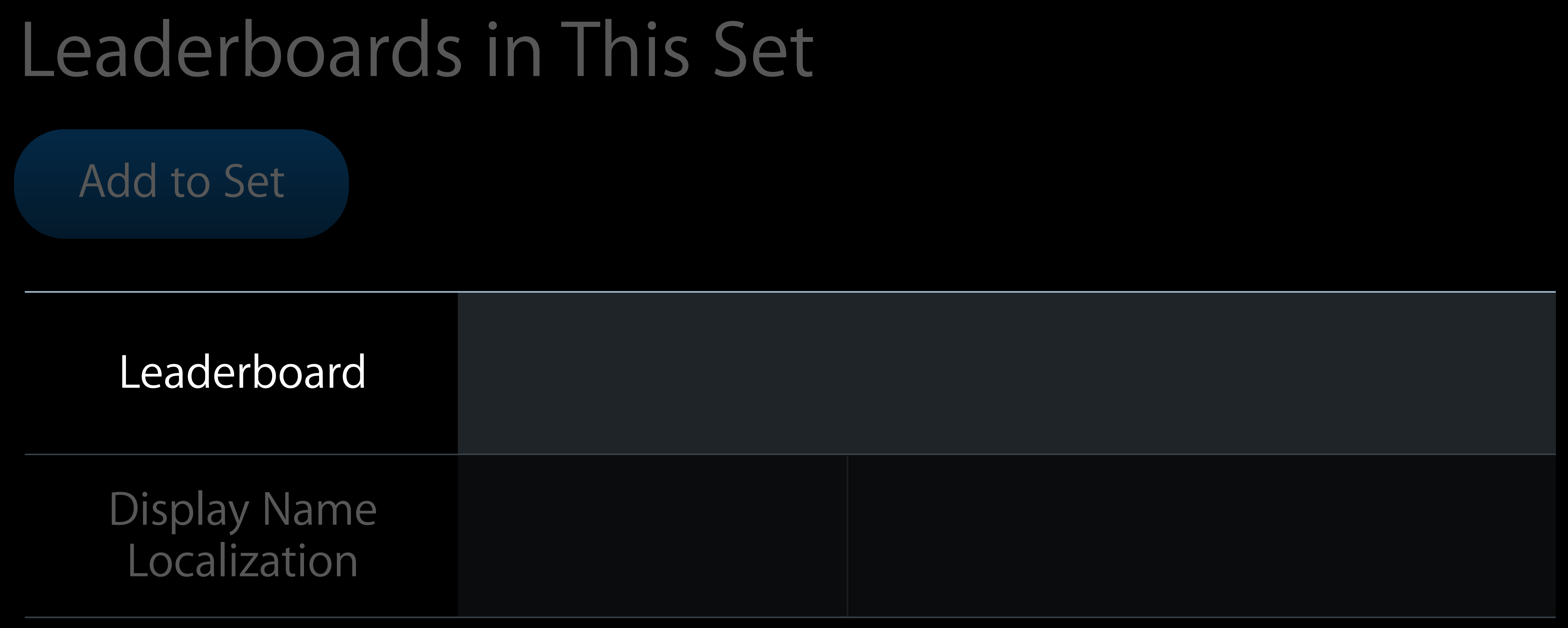
Leaderboards in This Set

Add to Set

Leaderboard		
Display Name Localization		

iTunes Connect Setup

Quick flyover

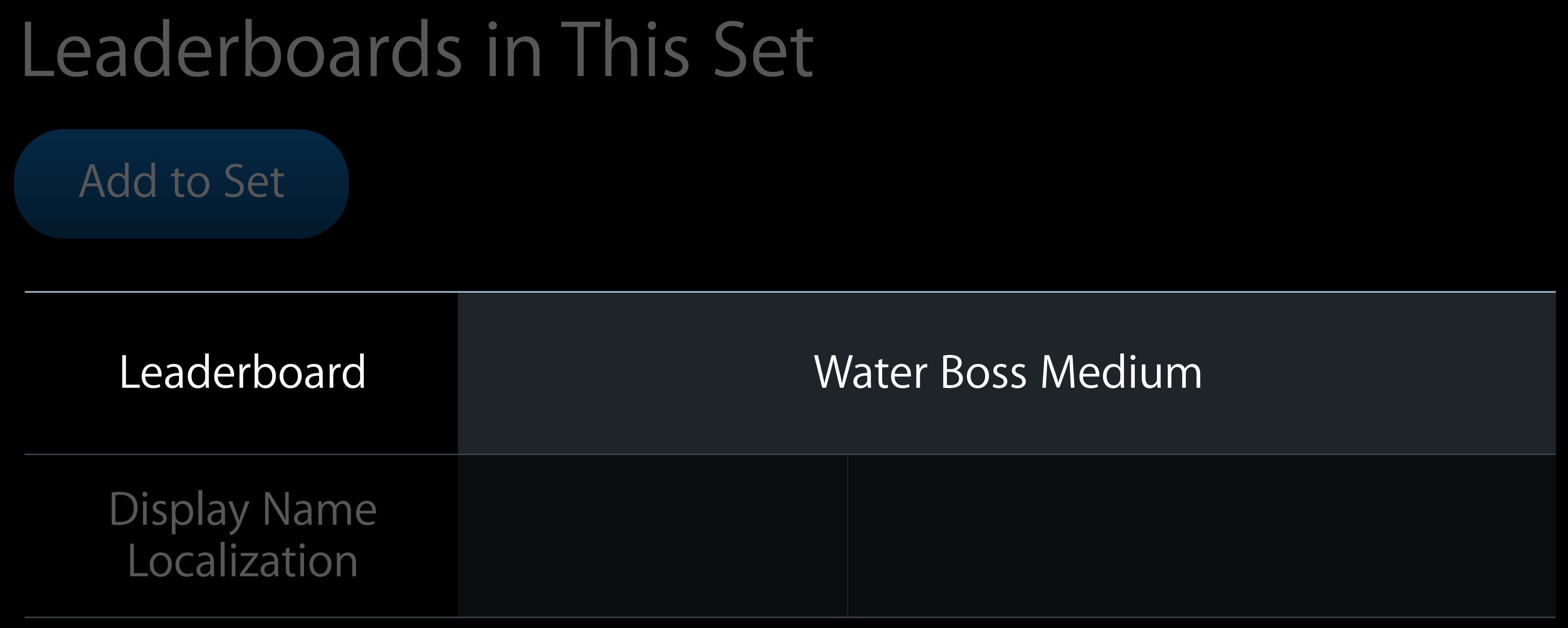


Leaderboards in This Set

Leaderboard		
Display Name Localization		

iTunes Connect Setup

Quick flyover

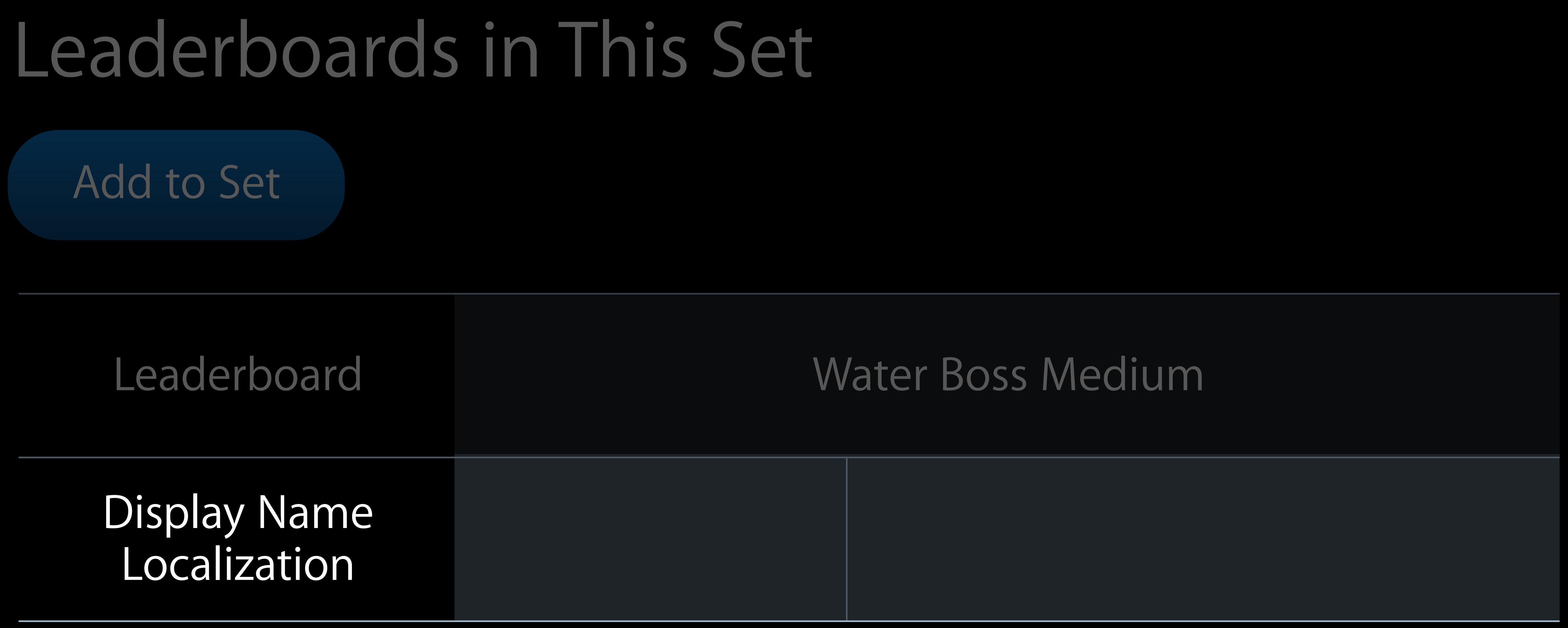


Leaderboards in This Set

Leaderboard	Water Boss Medium	
Display Name Localization		

iTunes Connect Setup

Quick flyover

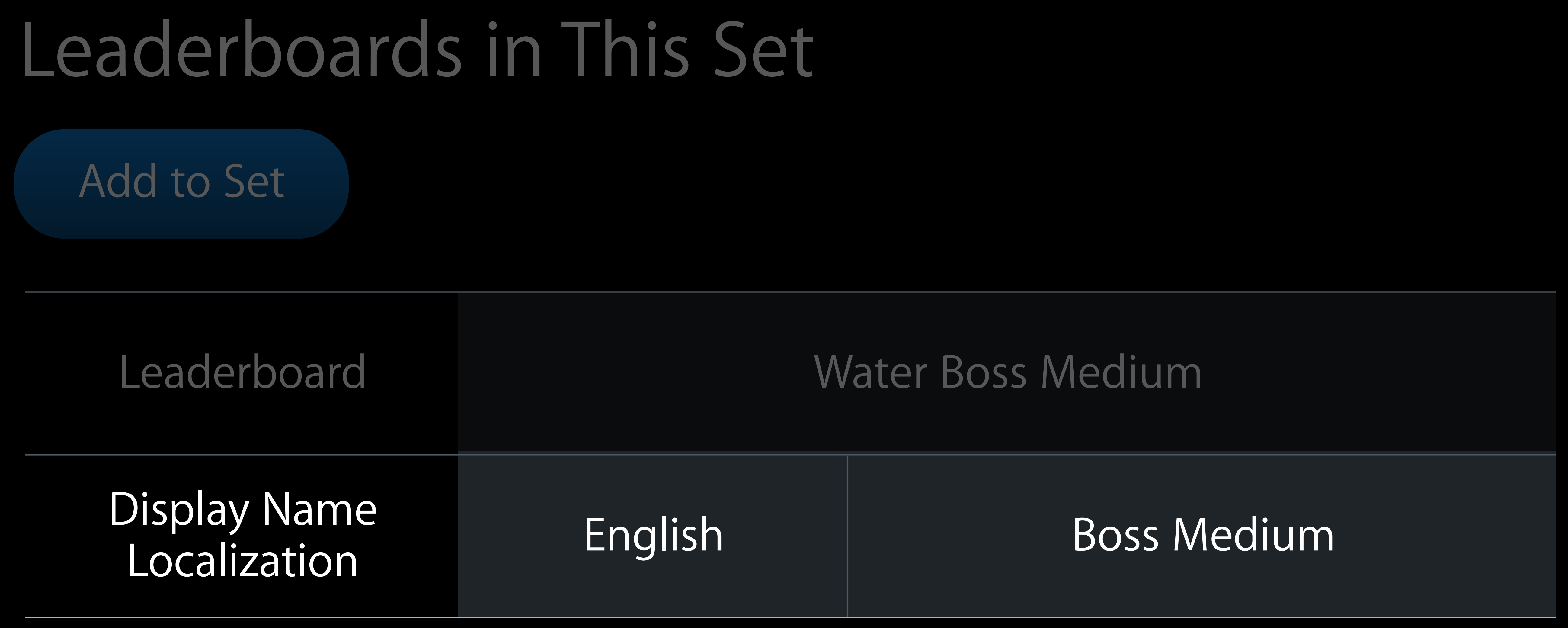


Leaderboards in This Set

Leaderboard	Water Boss Medium	
Display Name Localization		

iTunes Connect Setup

Quick flyover

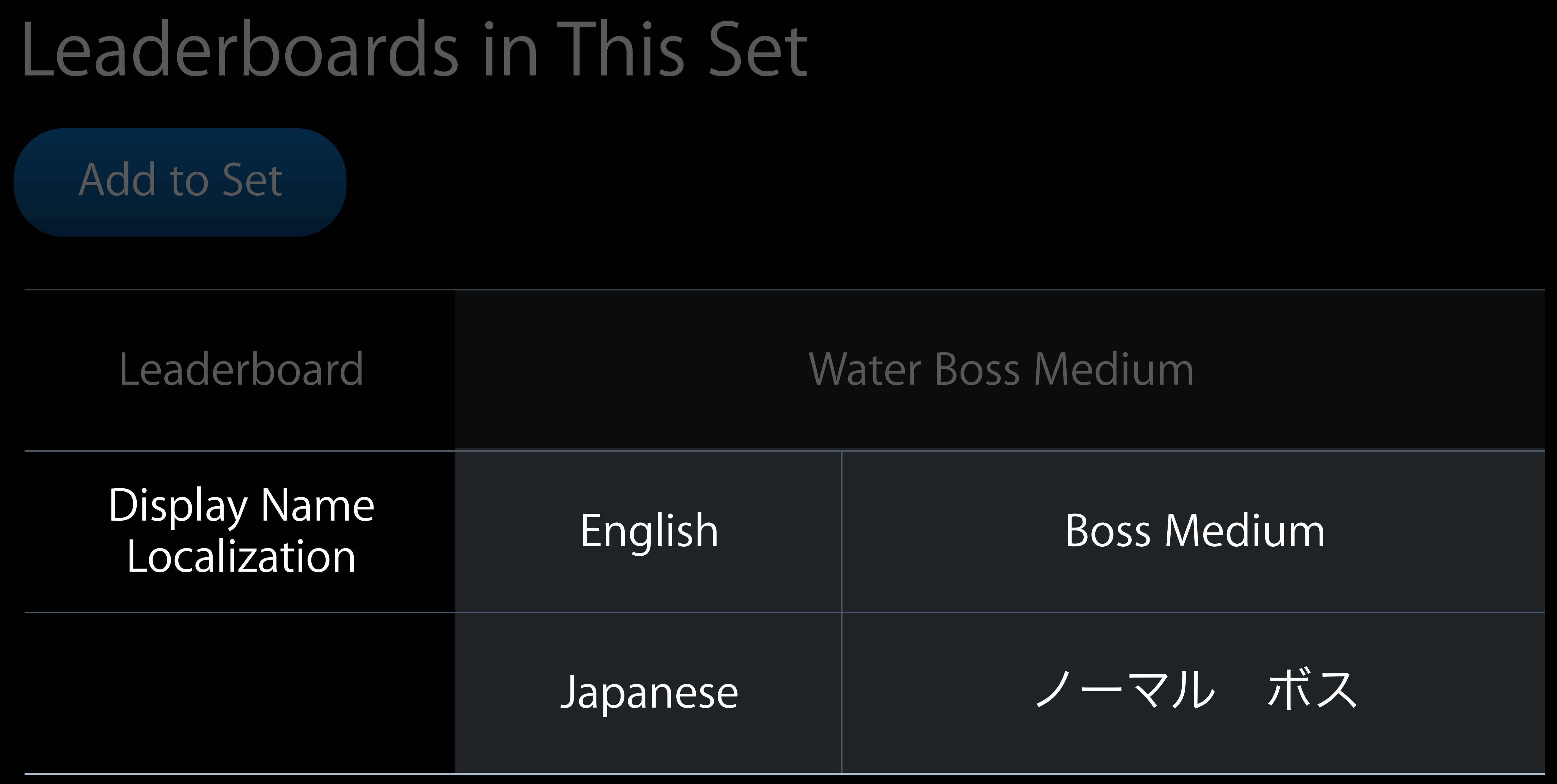


Leaderboards in This Set

Leaderboard	Water Boss Medium	
Display Name Localization	English	Boss Medium

iTunes Connect Setup

Quick flyover



Leaderboards in This Set

Leaderboard	Water Boss Medium	
Display Name Localization	English	Boss Medium
	Japanese	ノーマル　ボス

Using Leaderboard Sets

API

```
@interface GKLeaderboardSet

@property NSString *title;
@property NSString *identifier;
@property NSString *groupIdentifier;

@end
```

Using Leaderboard Sets

API

```
@interface GKLeaderboardSet

@property NSString *title;
@property NSString *identifier;
@property NSString *groupIdentifier;

@end
```

Using Leaderboard Sets

API

```
@interface GKLeaderboardSet

@property NSString *title;
@property NSString *identifier;
@property NSString *groupIdentifier;

@end
```

Using Leaderboard Sets

API

```
@interface GKLeaderboardSet

@property NSString *title;
@property NSString *identifier;
@property NSString *groupIdentifier;

@end
```

Using Leaderboard Sets

Loading Leaderboard Sets

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {
    // ...
}];
```

Using Leaderboard Sets

Loading Leaderboard Sets

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {
    // ...
}];
```

Using Leaderboard Sets

Loading Leaderboards from Sets

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadLeaderboardsWithCompletionHandler: ^(NSArray
*leaderboards, NSError *error) {
        // Use loaded GKLeaderboard objects just like normal
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboards from Sets

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadLeaderboardsWithCompletionHandler: ^(NSArray
*leaderboards, NSError *error) {
        // Use loaded GKLeaderboard objects just like normal
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboards from Sets

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadLeaderboardsWithCompletionHandler: ^(NSArray
*leaderboards, NSError *error) {
        // Use loaded GKLeaderboard objects just like normal
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboards from Sets

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadLeaderboardsWithCompletionHandler: ^(NSArray
*leaderboards, NSError *error) {
        // Use loaded GKLeaderboard objects just like normal
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboards from Sets

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadLeaderboardsWithCompletionHandler: ^(NSArray
*leaderboards, NSError *error) {
        // Use loaded GKLeaderboard objects just like normal
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboard Set Images

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadImageWithCompletionHandler: ^(GKImage *image, NSError
*error) {
        // ...
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboard Set Images

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadImageWithCompletionHandler: ^(GKImage *image, NSError
*error) {
        // ...
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboard Set Images

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadImageWithCompletionHandler: ^(GKImage *image, NSError
*error) {
        // ...
    }];

}];
```

Using Leaderboard Sets

Loading Leaderboard Set Images

```
// GKLeaderboardSet objects must be loaded before use
[GKLeaderboardSet loadLeaderboardSetsWithCompletionHandler: ^(NSArray
*leaderboardSets, NSError *error) {

    GKLeaderboardSet *myLoadedSet = [leaderboardSets firstObject];

    [myLoadedSet loadImageWithCompletionHandler: ^(GKImage *image, NSError
*error) {
        // ...
    }];

}];
```

Leaderboard Sets

- Leaderboards tagged to sets
- One leaderboard to many sets
- Once enabled, all leaderboards must belong to a set

Most Recent Score Leaderboards

Most Recent Score Leaderboards

- Alternative to Best Score Leaderboards
- Ranking system entirely customizable
  - Win/Loss
  - Power Ranking
  - Batting Average

iTunes Connect Setup

Quick flyover

Leaderboard Information

Leaderboard ID	`com.studioName.myGame.leaderboardID`			
Score Submission Type	Best Score	○	Most Recent Score	○
Sort Order	Low to High	○	High to Low	○
Score Range	From	□	To	□

iTunes Connect Setup

Quick flyover

Leaderboard Information

Leaderboard ID	`com.studioName.myGame.leaderboardID`			
Score Submission Type	Best Score	○	Most Recent Score	○
Sort Order	Low to High	○	High to Low	○
Score Range	From	□	To	□

iTunes Connect Setup

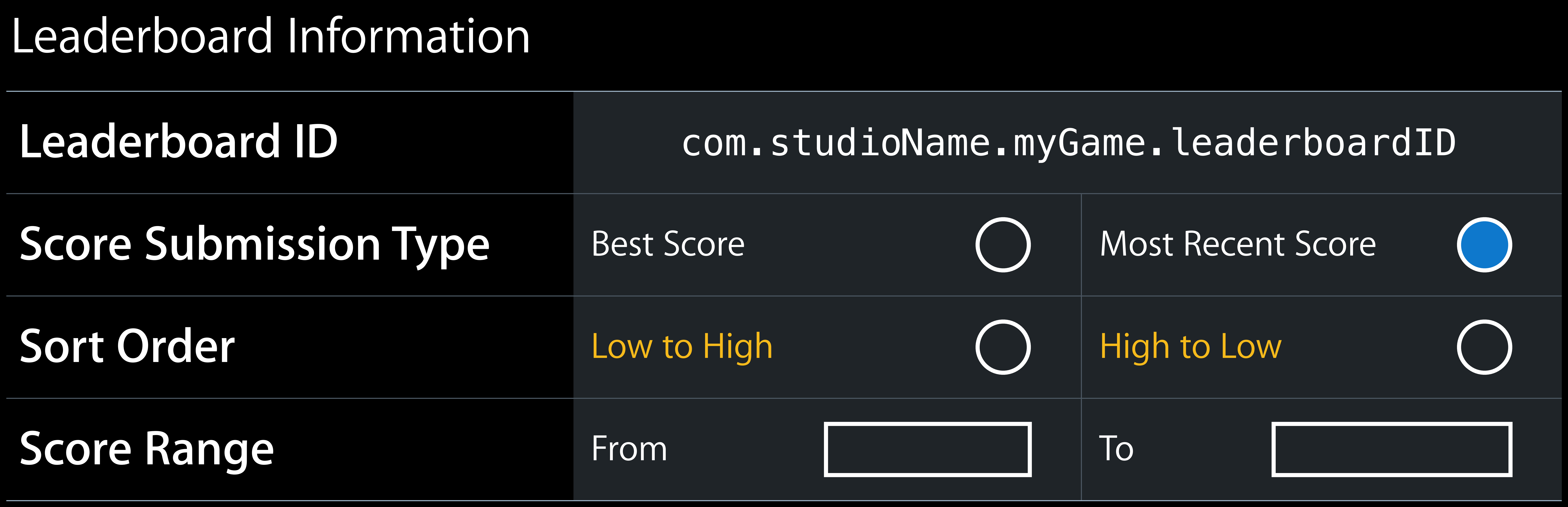
Quick flyover

Leaderboard Information

Leaderboard ID	com.studioName.myGame.leaderboardID	
Score Submission Type	Best Score ○	Most Recent Score ●
Sort Order	Low to High ○	High to Low ○
Score Range	From □	To □

iTunes Connect Setup

Quick flyover



Leaderboard Information

Leaderboard ID	`com.studioName.myGame.leaderboardID`	
Score Submission Type	Best Score ○	Most Recent Score ●
Sort Order	Low to High ○	High to Low ○
Score Range	From □	To □

iTunes Connect Setup

Quick flyover

Leaderboard Information

Leaderboard ID	`com.studioName.myGame.leaderboardID`			
Score Submission Type	Best Score	○	Most Recent Score	●
Sort Order	Low to High	●	High to Low	○
Score Range	From	□	To	□

iTunes Connect Setup

Quick flyover

Leaderboard Information

Leaderboard ID	`com.studioName.myGame.leaderboardID`	
Score Submission Type	Best Score ○	Most Recent Score ●
Sort Order	Low to High ○	High to Low ●
Score Range	From	To

Using Most Recent Score Leaderboards

```
// load scores for both players
GKLeaderboard *leaderboard = [[GKLeaderboard alloc] initWithPlayers:
@[localPlayer, otherPlayer]];

leaderboard.identifier = recentScoreLeaderboardID;

[leaderboard loadScoresWithCompletionHandler: ^(NSArray *scores, NSError *)
{
    GKScore *localScore, *otherScore = // get from scores array
    int64_t localValue = localScore.value;
    int64_t otherValue = otherScore.value;

    // calculate new power ranking based on both scores and match result
    int64_t newLocalValue = [self powerRankingForLocal:localValue
    other:otherValue didWinMatch:won];

    // continued...
```

Using Most Recent Score Leaderboards

```
// load scores for both players
GKLeaderboard *leaderboard = [[GKLeaderboard alloc] initWithPlayers:
@[localPlayer, otherPlayer]];

leaderboard.identifier = recentScoreLeaderboardID;

[leaderboard loadScoresWithCompletionHandler: ^(NSArray *scores, NSError *)
{
    GKScore *localScore, *otherScore = // get from scores array
    int64_t localValue = localScore.value;
    int64_t otherValue = otherScore.value;

    // calculate new power ranking based on both scores and match result
    int64_t newLocalValue = [self powerRankingForLocal:localValue
    other:otherValue didWinMatch:won];

    // continued...
```

Using Most Recent Score Leaderboards

```
// load scores for both players
GKLeaderboard *leaderboard = [[GKLeaderboard alloc] initWithPlayers:
@[localPlayer, otherPlayer]];

leaderboard.identifier = recentScoreLeaderboardID;

[leaderboard loadScoresWithCompletionHandler: ^(NSArray *scores, NSError *)
{
    GKScore *localScore, *otherScore = // get from scores array
    int64_t localValue = localScore.value;
    int64_t otherValue = otherScore.value;

    // calculate new power ranking based on both scores and match result
    int64_t newLocalValue = [self powerRankingForLocal:localValue
    other:otherValue didWinMatch:won];

    // continued...
```

Using Most Recent Score Leaderboards

```
// load scores for both players
GKLeaderboard *leaderboard = [[GKLeaderboard alloc] initWithPlayers:
@[localPlayer, otherPlayer]];

leaderboard.identifier = recentScoreLeaderboardID;

[leaderboard loadScoresWithCompletionHandler: ^(NSArray *scores, NSError *)
{
    GKScore *localScore, *otherScore = // get from scores array
    int64_t localValue = localScore.value;
    int64_t otherValue = otherScore.value;

    // calculate new power ranking based on both scores and match result
    int64_t newLocalValue = [self powerRankingForLocal:localValue
    other:otherValue didWinMatch:won];

    // continued...
```

Using Most Recent Score Leaderboards

```
// load scores for both players
GKLeaderboard *leaderboard = [[GKLeaderboard alloc] initWithPlayers:
@[localPlayer, otherPlayer]];

leaderboard.identifier = recentScoreLeaderboardID;

[leaderboard loadScoresWithCompletionHandler: ^(NSArray *scores, NSError *)
{
    GKScore *localScore, *otherScore = // get from scores array
    int64_t localValue = localScore.value;
    int64_t otherValue = otherScore.value;

    // calculate new power ranking based on both scores and match result
    int64_t newLocalValue = [self powerRankingForLocal:localValue
    other:otherValue didWinMatch:won];

    // continued...
```

Using Most Recent Score Leaderboards

```
// load scores for both players
GKLeaderboard *leaderboard = [[GKLeaderboard alloc] initWithPlayers:
@[localPlayer, otherPlayer]];

leaderboard.identifier = recentScoreLeaderboardID;

[leaderboard loadScoresWithCompletionHandler: ^(NSArray *scores, NSError *)
{
    GKScore *localScore, *otherScore = // get from scores array
    int64_t localValue = localScore.value;
    int64_t otherValue = otherScore.value;

    // calculate new power ranking based on both scores and match result
    int64_t newLocalValue = [self powerRankingForLocal:localValue
    other:otherValue didWinMatch:won];

    // continued...
```

Using Most Recent Score Leaderboards

```
    // submit
    GKScore *score = [[GKScore alloc] initWithLeaderboardIdentifier:
                                     recentScoreLeaderboardID];
    score.value = newLocalValue;
    [GKScore reportScores: @[score] withCompletionHandler: nil];

}];
```

Using Most Recent Score Leaderboards

```
    // submit
    GKScore *score = [[GKScore alloc] initWithLeaderboardIdentifier:
                                      recentScoreLeaderboardID];
    score.value = newLocalValue;
    [GKScore reportScores: @[score] withCompletionHandler: nil];

}];
```

Using Most Recent Score Leaderboards

```
    // submit
    GKScore *score = [[GKScore alloc] initWithLeaderboardIdentifier:
                                     recentScoreLeaderboardID];
    score.value = newLocalValue;
    [GKScore reportScores: @[score] withCompletionHandler: nil];

}];
```

Using Most Recent Score Leaderboards

```
    // submit
    GKScore *score = [[GKScore alloc] initWithLeaderboardIdentifier:
                                      recentScoreLeaderboardID];
    score.value = newLocalValue;
    [GKScore reportScores: @[score] withCompletionHandler: nil];

}];
```

GKLeaderboard API Changes

```
-[GKLeaderboard category]

-[GKLeaderboard identifier]

+[GKLeaderboard setDefaultLeaderboard: withCompletionHandler:]

-[GKLocalPlayer setDefaultLeaderboardIdentifier: completionHandler:]
```

GKLeaderboard API Changes

```
-[GKLeaderboard category]
```
✗

```
-[GKLeaderboard identifier]

+[GKLeaderboard setDefaultLeaderboard: withCompletionHandler:]

-[GKLocalPlayer setDefaultLeaderboardIdentifier: completionHandler:]
```

GKLeaderboard API Changes

`-[GKLeaderboard category]` ✗

`-[GKLeaderboard identifier]` ✓

```
+[GKLeaderboard setDefaultLeaderboard: withCompletionHandler:]

-[GKLocalPlayer setDefaultLeaderboardIdentifier: completionHandler:]
```

GKLeaderboard API Changes

-[GKLeaderboard category] ✗

-[GKLeaderboard identifier] ✓

+[GKLeaderboard setDefaultLeaderboard: withCompletionHandler:] ✗

-[GKLocalPlayer setDefaultLeaderboardIdentifier: completionHandler:]

GKLeaderboard API Changes

`-[GKLeaderboard category]` ✗

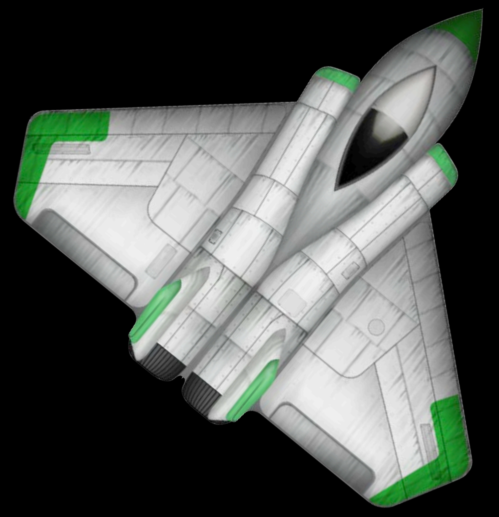
`-[GKLeaderboard identifier]` ✓

`+[GKLeaderboard setDefaultLeaderboard: withCompletionHandler:]` ✗

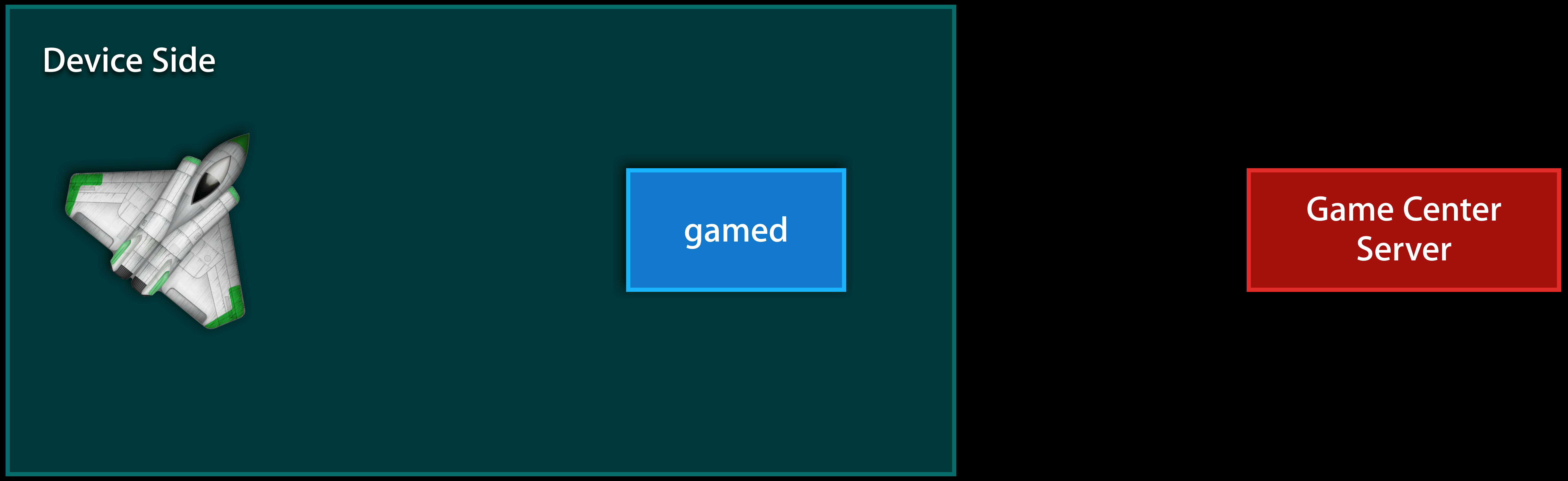
`-[GKLocalPlayer setDefaultLeaderboardIdentifier: completionHandler:]` ✓

How Submission Works

How Submission Works



How Submission Works

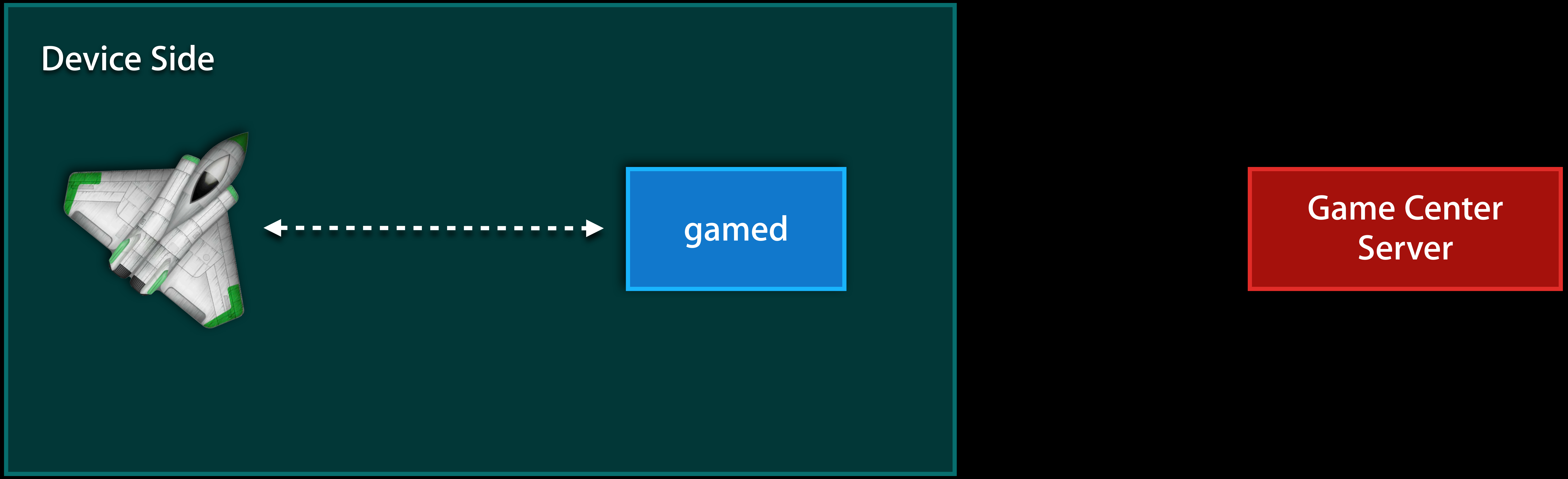


Device Side

gamed

Game Center Server

How Submission Works

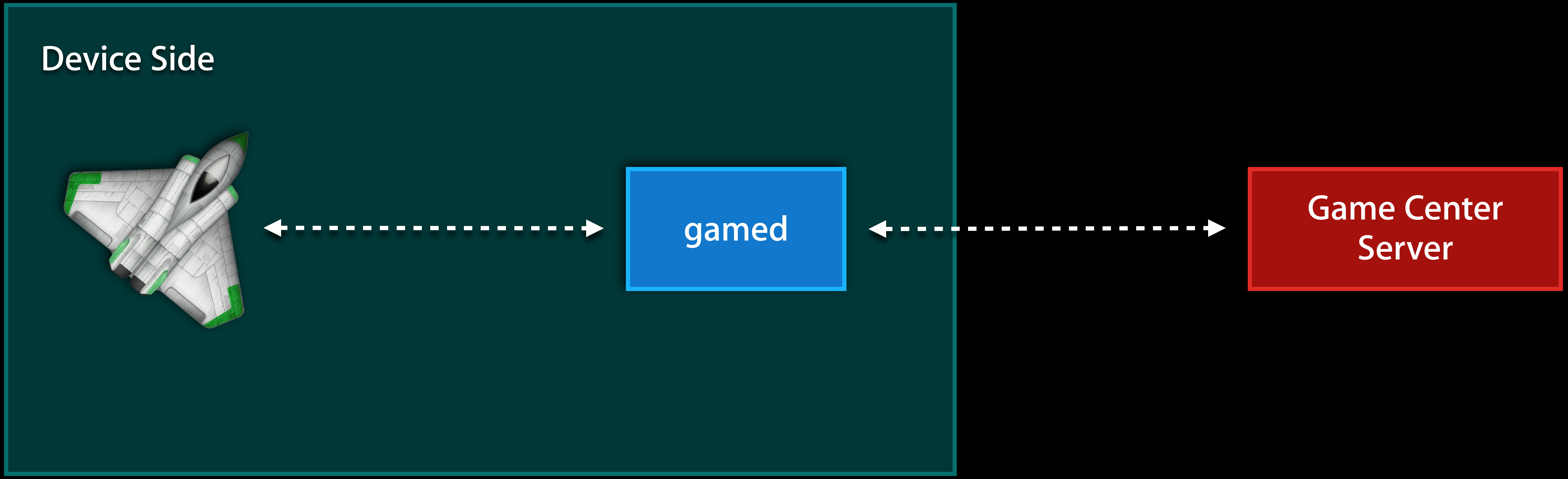


Device Side

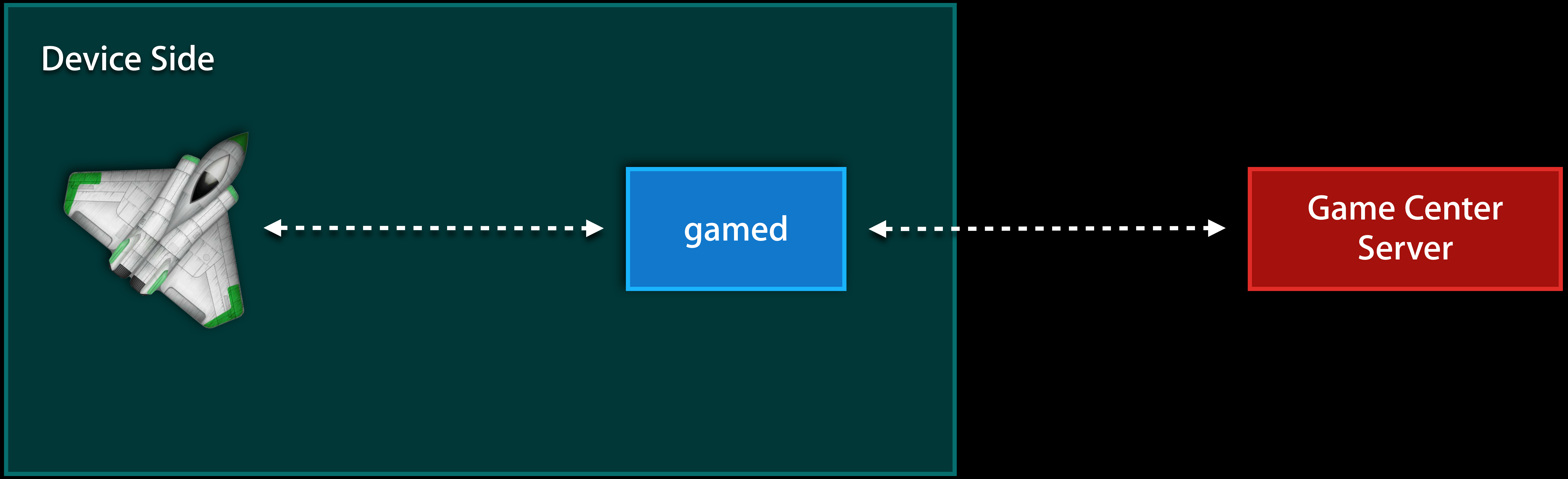
gamed

Game Center Server

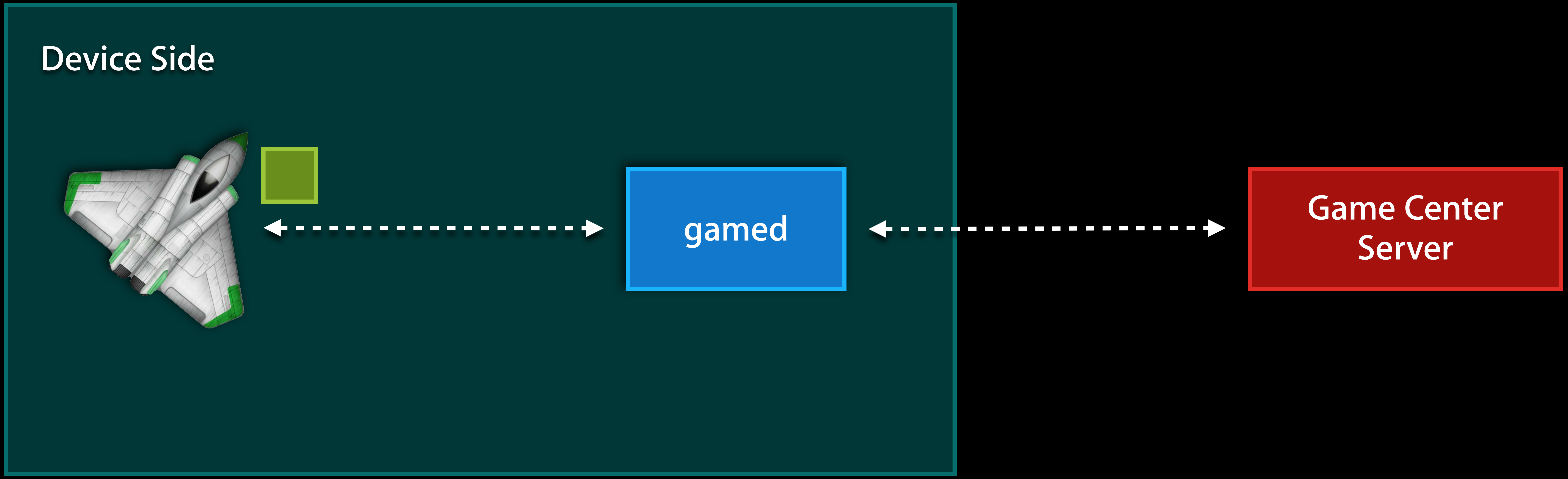
How Submission Works



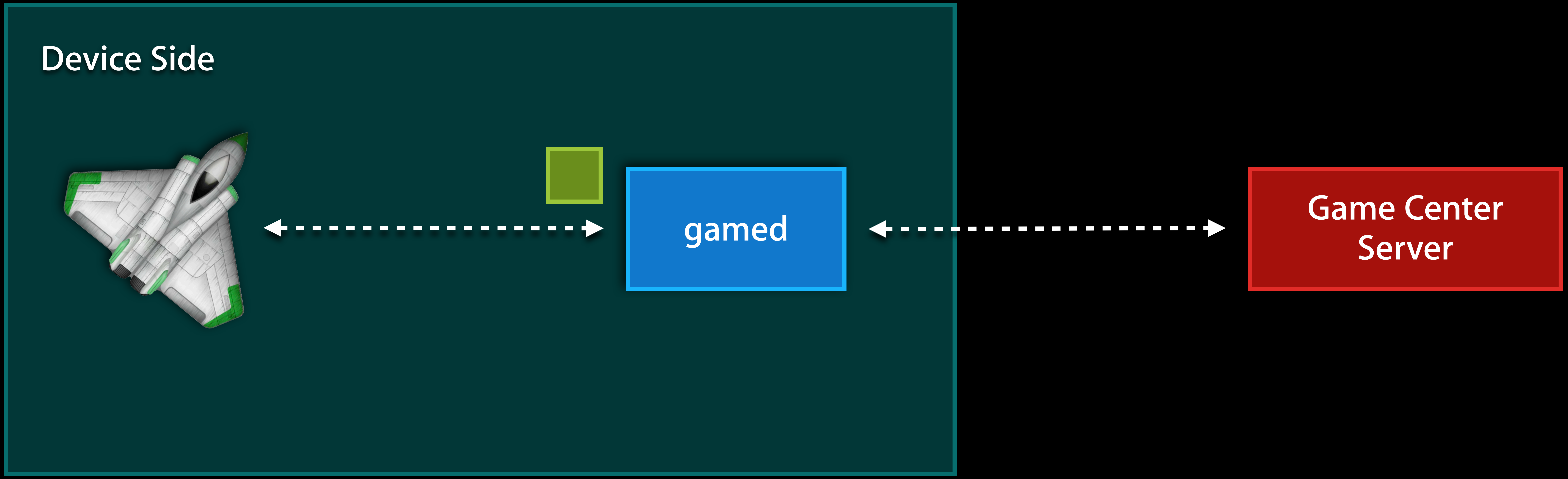
How Submission Works



How Submission Works



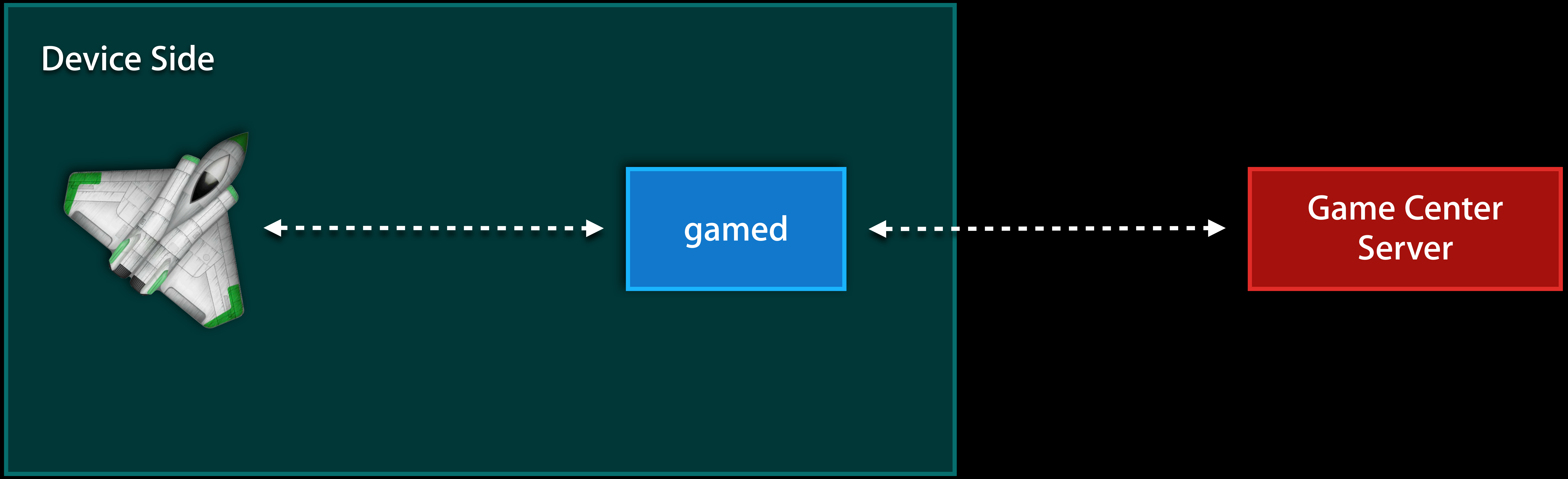
How Submission Works


Device Side

gamed

Game Center Server

How Submission Works

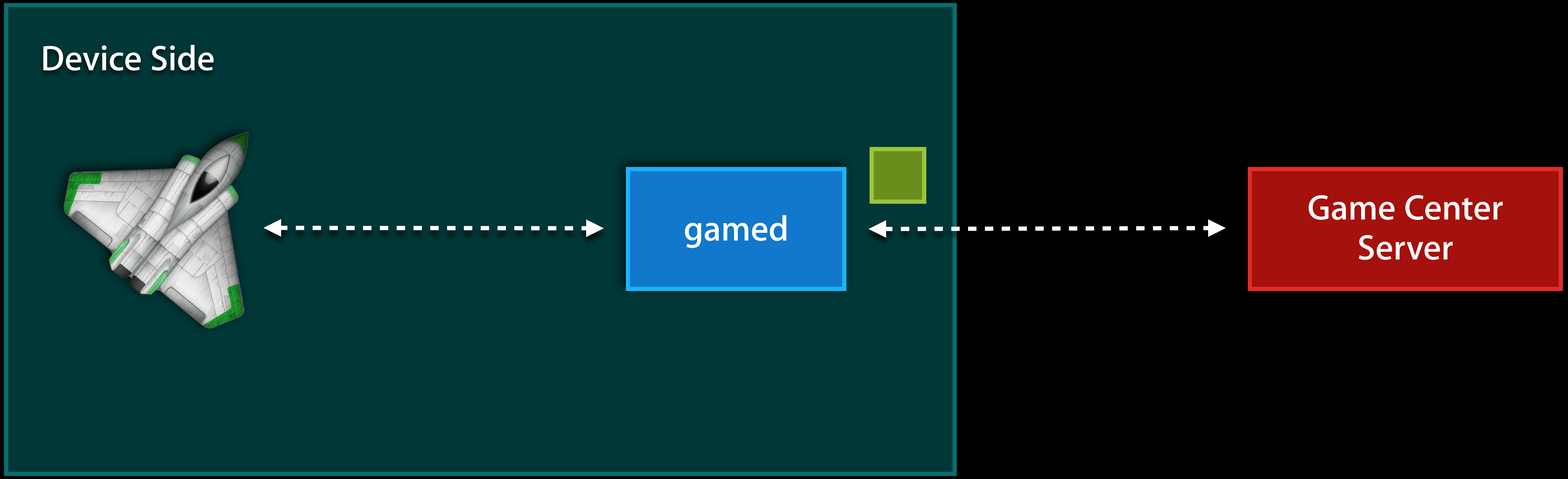


Device Side

gamed

Game Center Server

How Submission Works

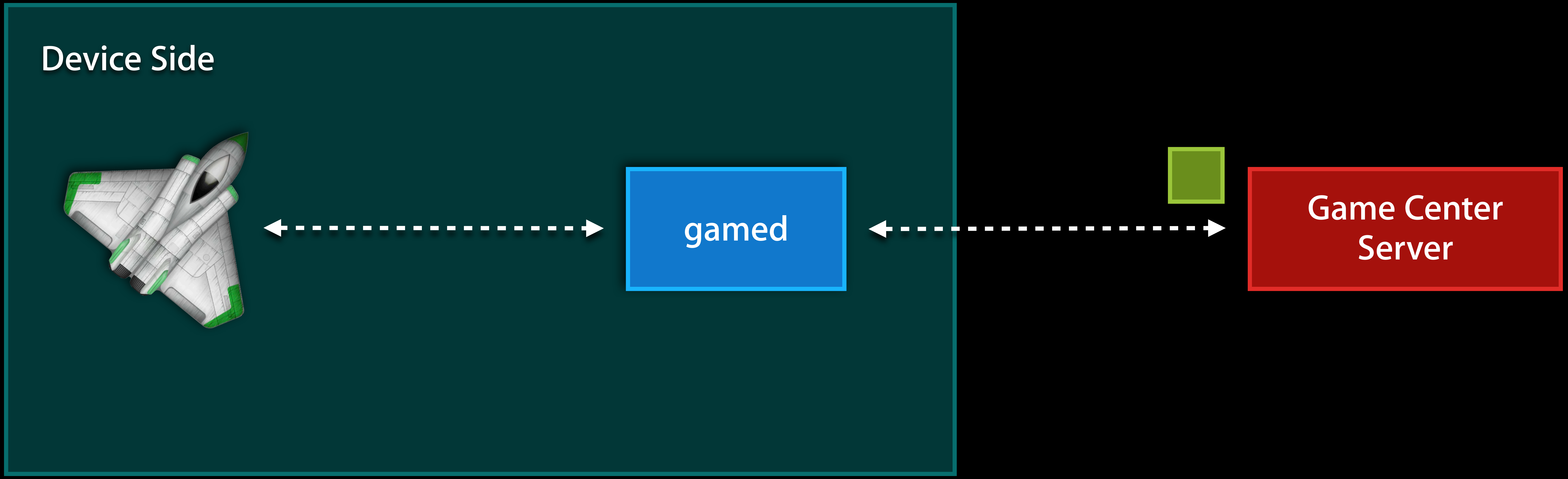


Device Side

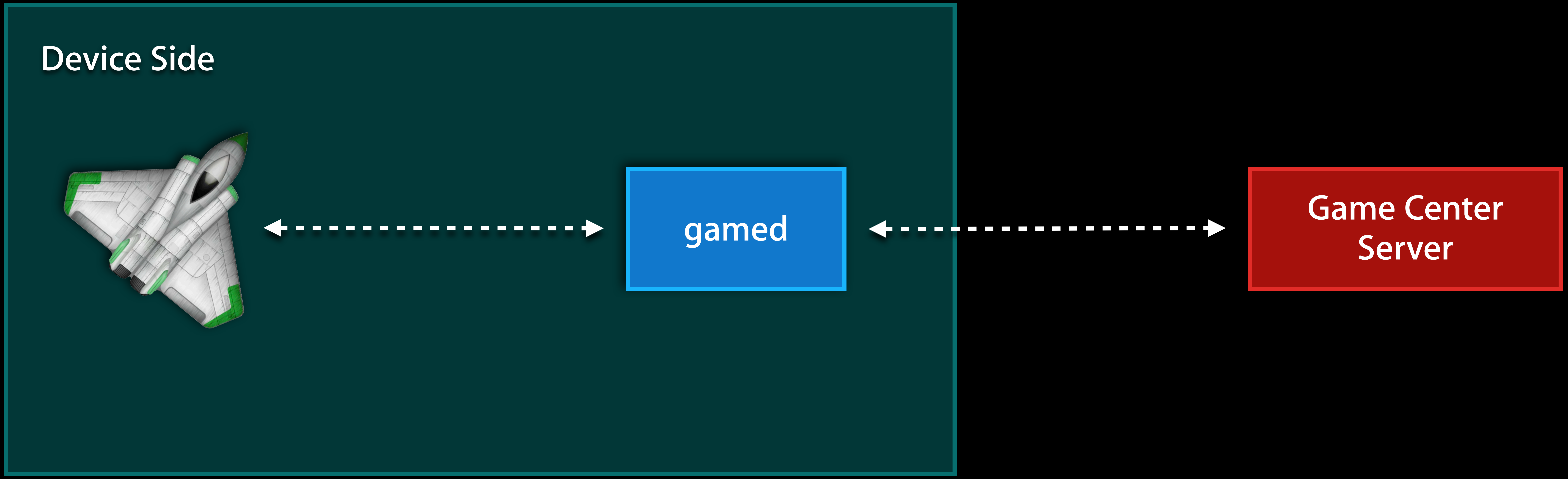
gamed

Game Center Server

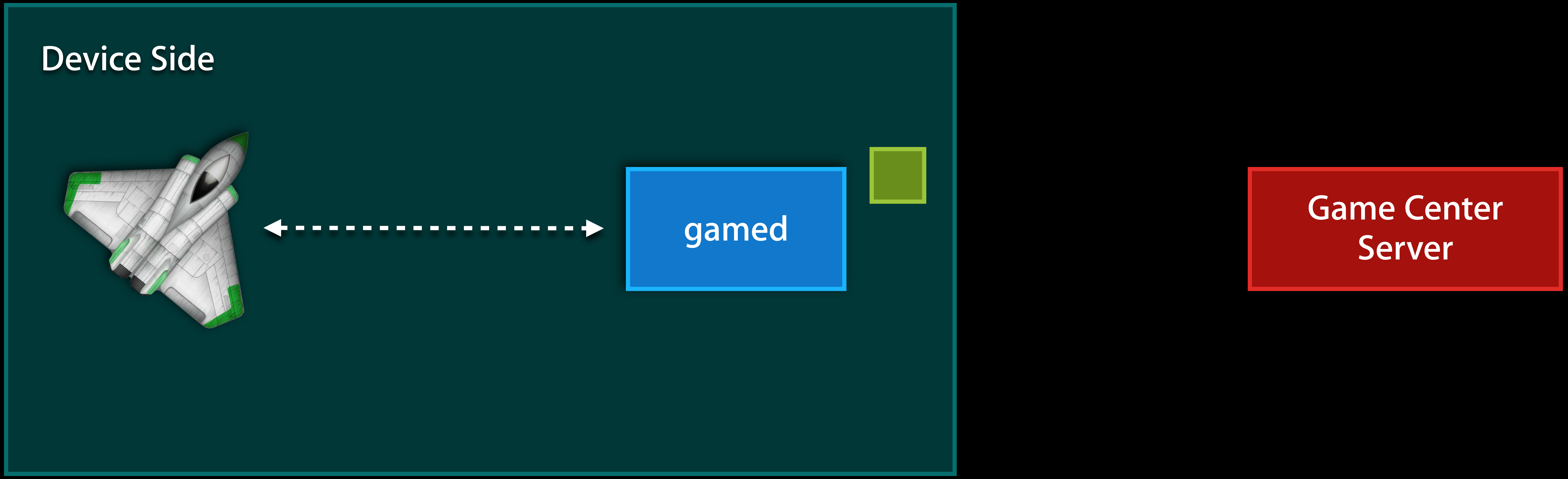
How Submission Works



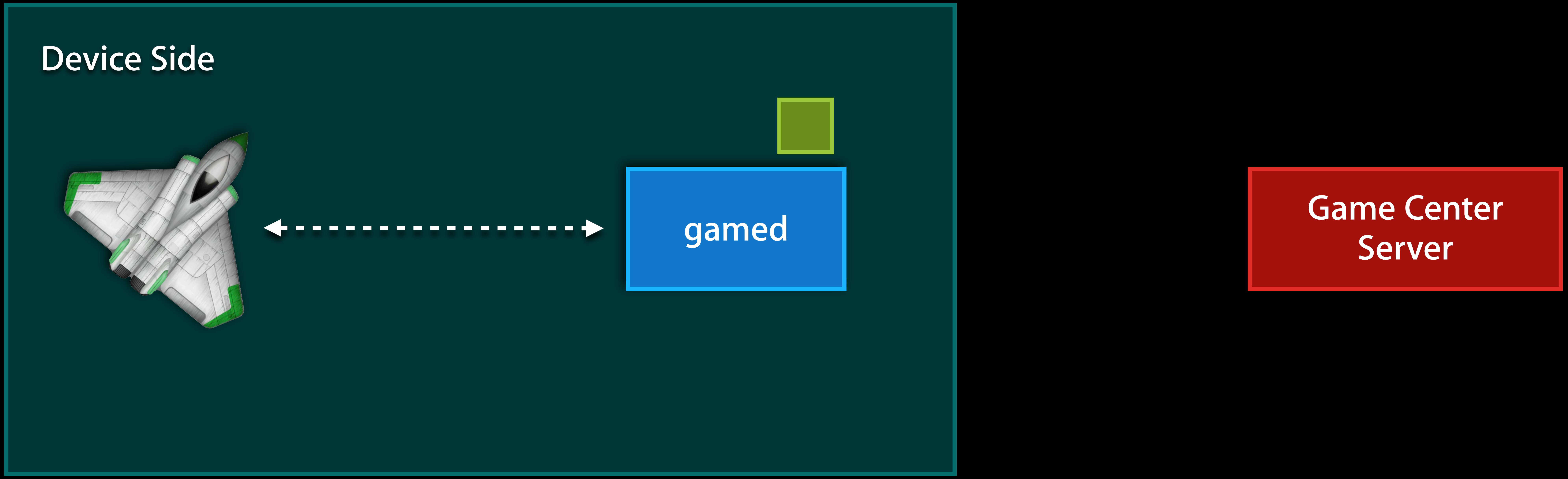
How Submission Works



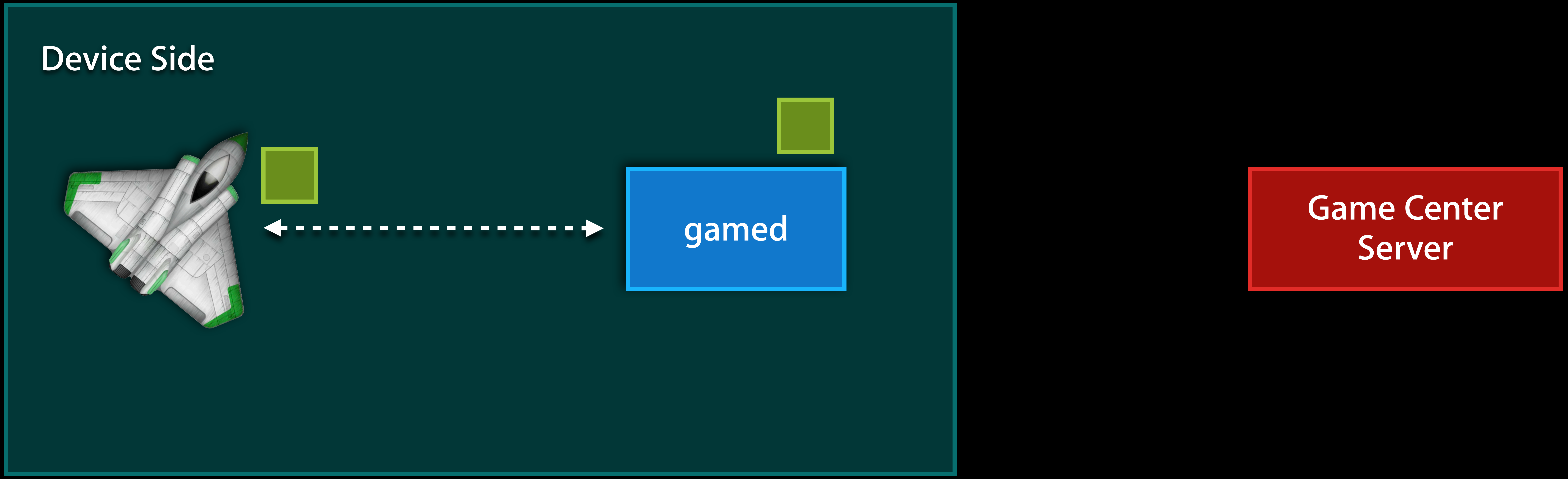
How Submission Works



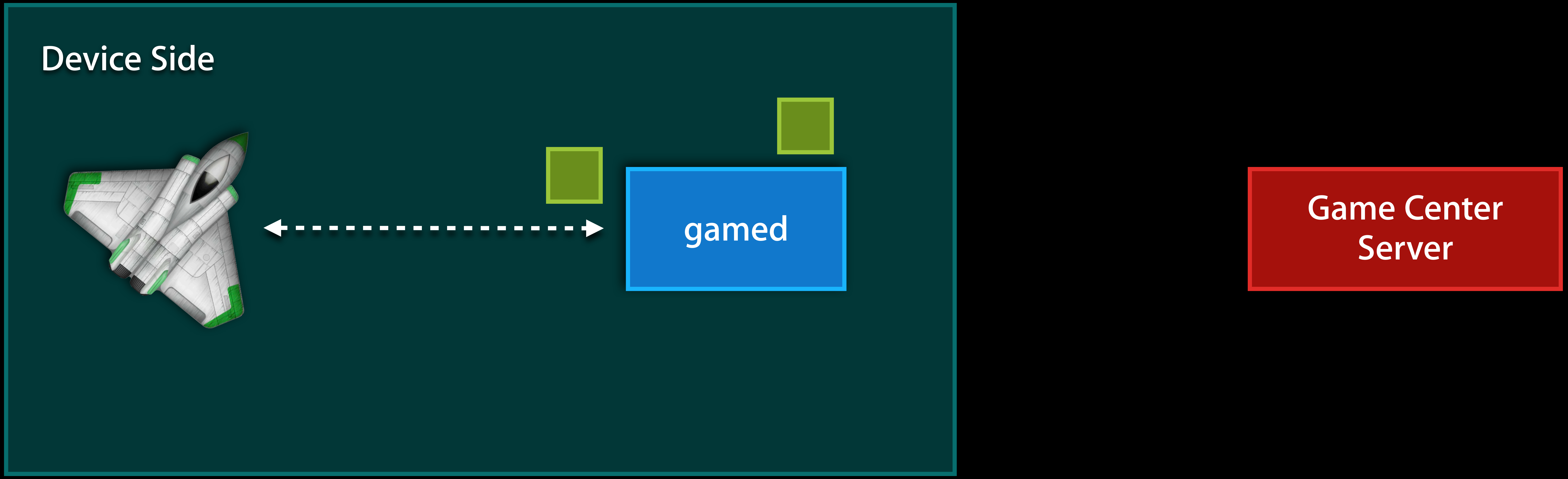
How Submission Works



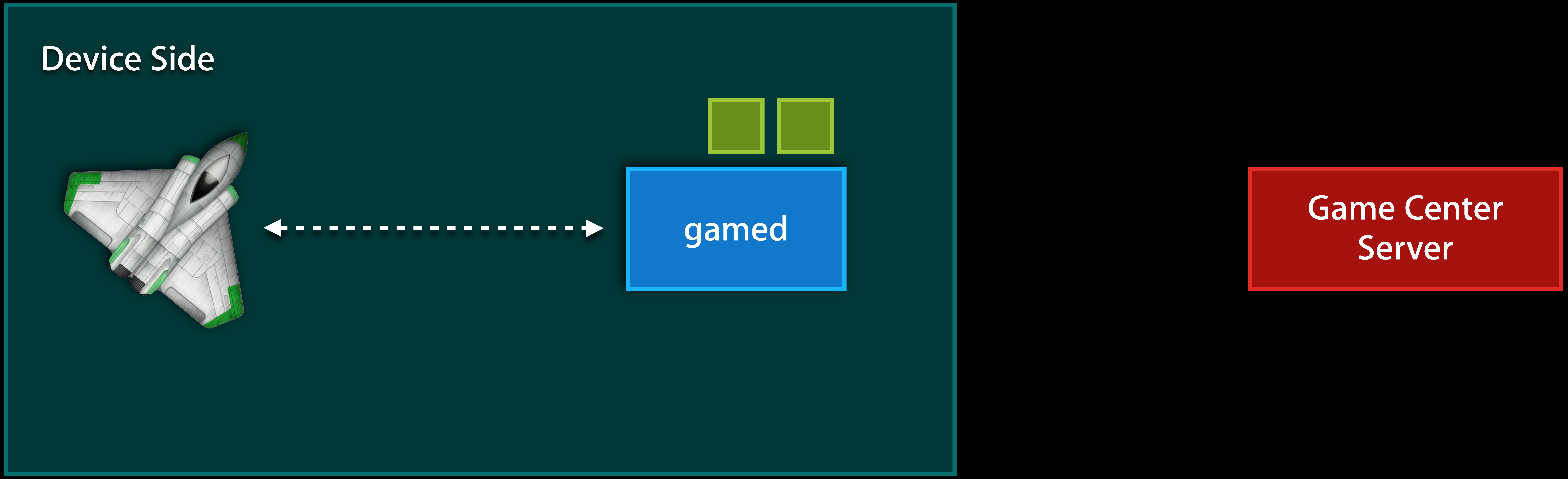
How Submission Works



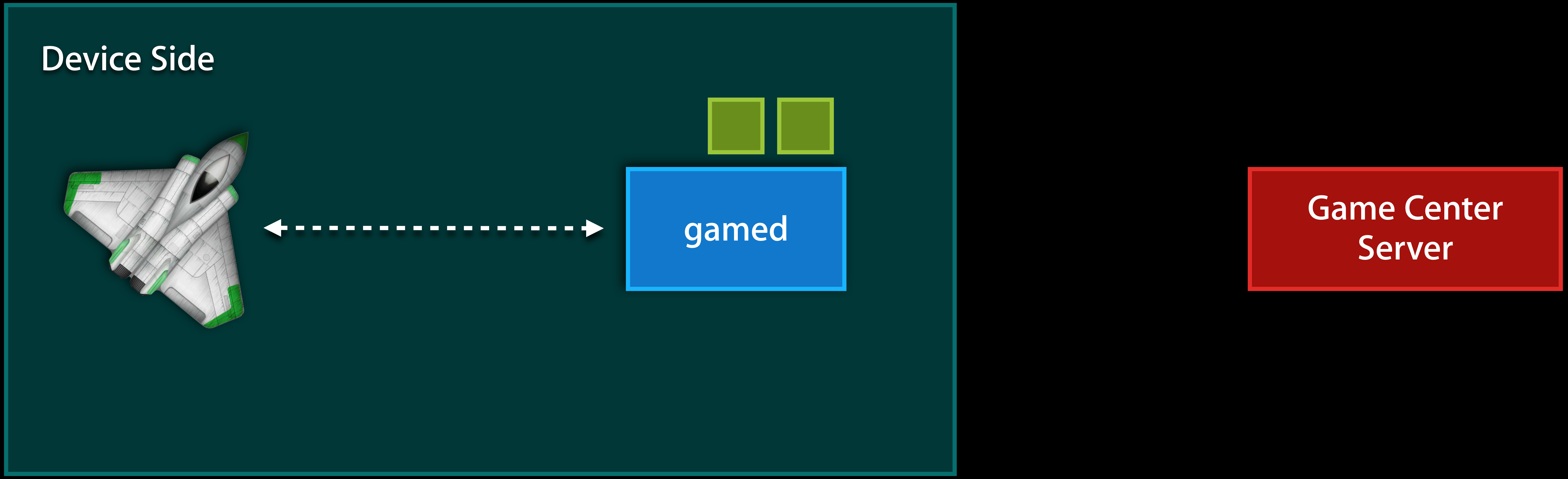
How Submission Works



How Submission Works



How Submission Works

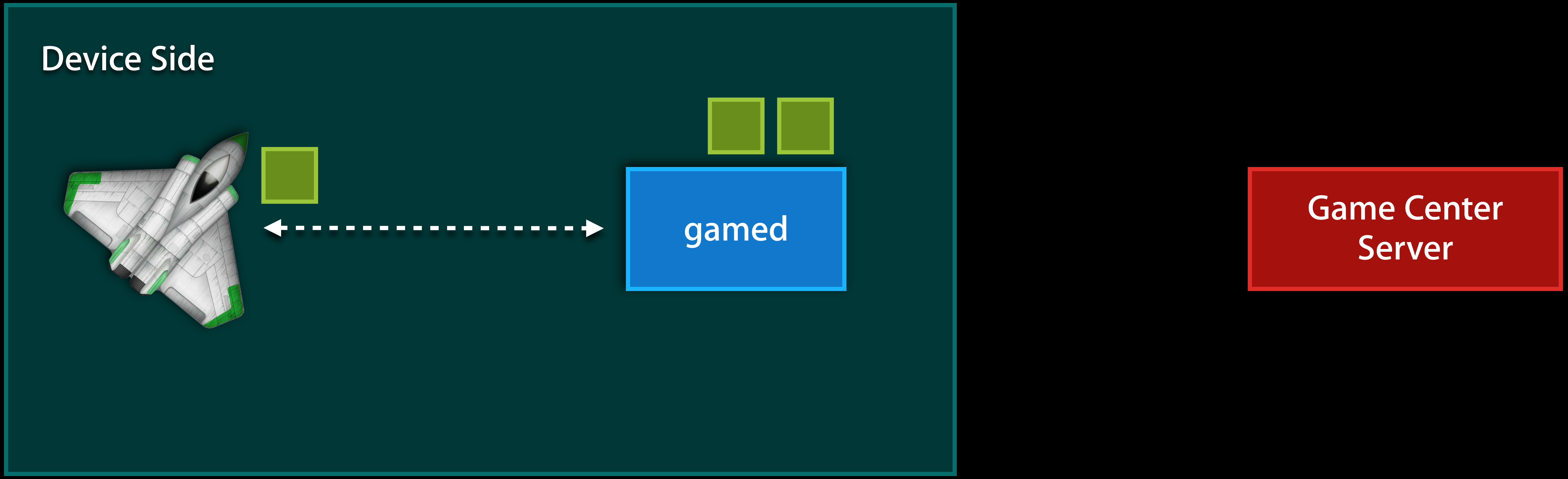


Device Side

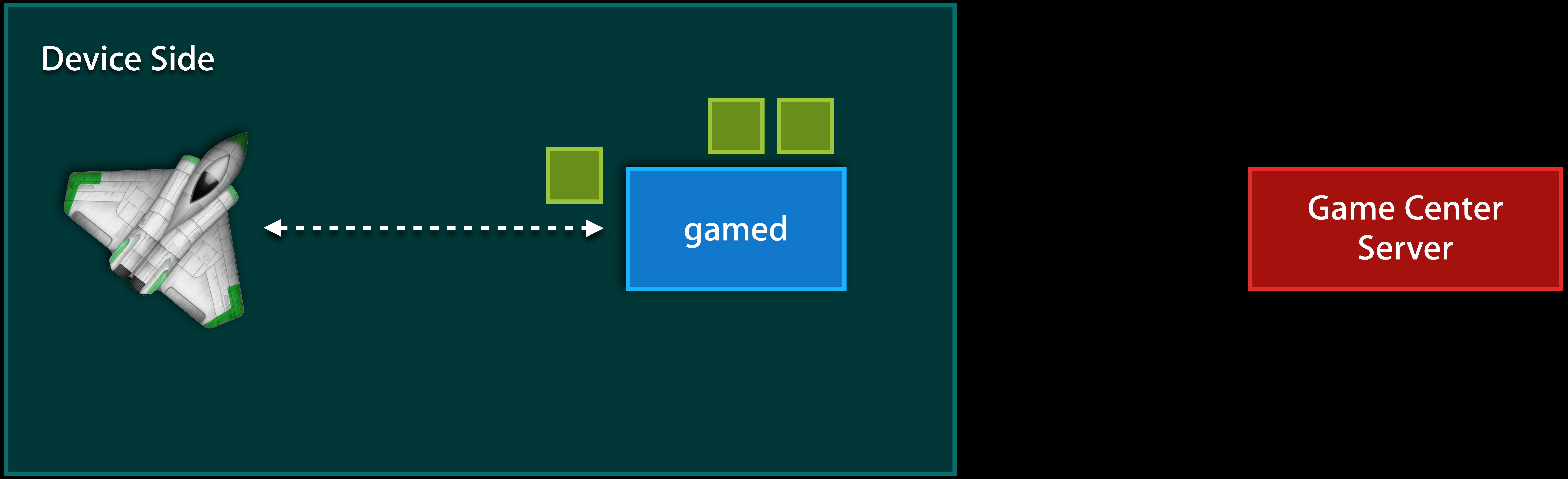
gamed

Game Center Server

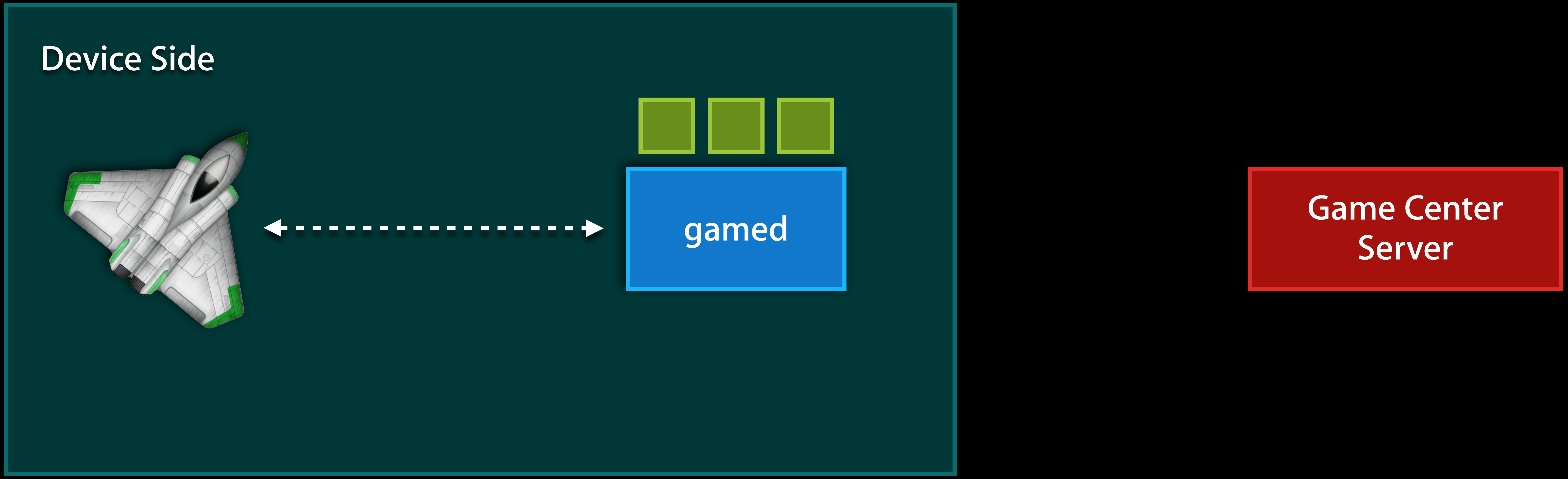
How Submission Works



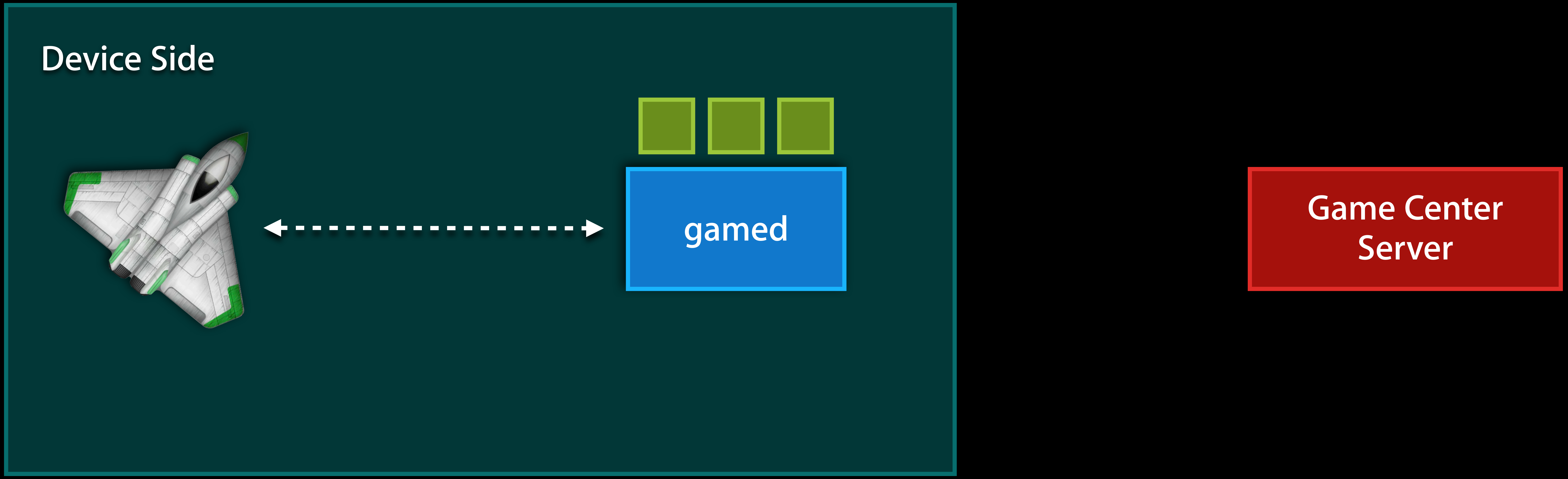
How Submission Works



How Submission Works



How Submission Works

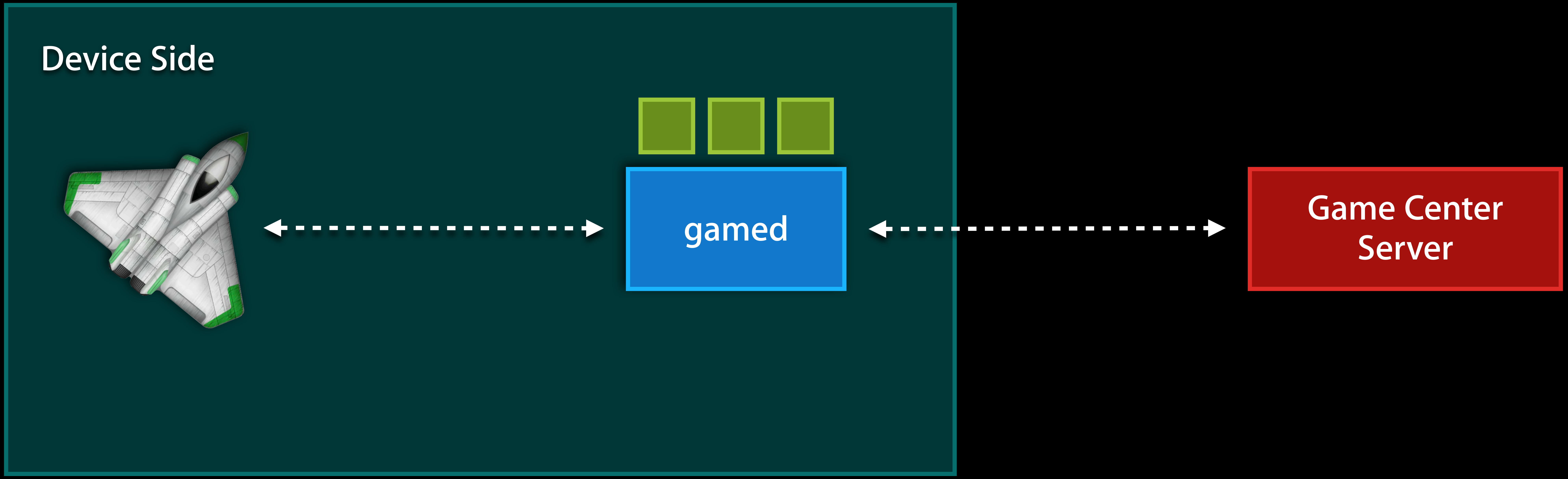


Device Side

gamed

Game Center Server

How Submission Works

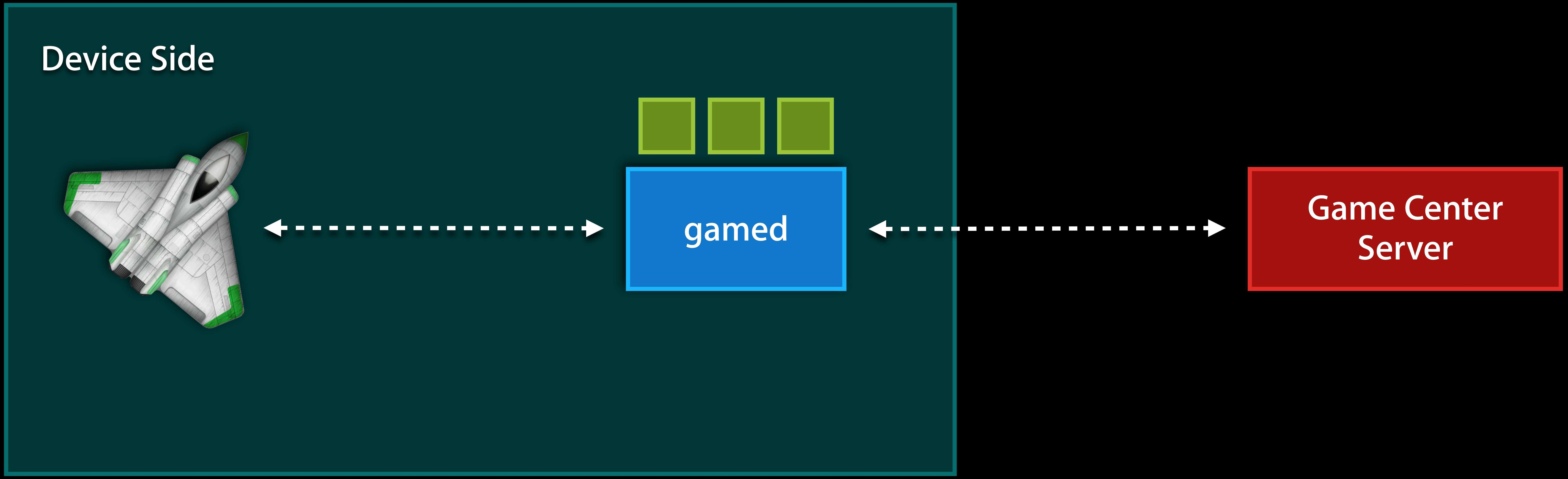


Device Side

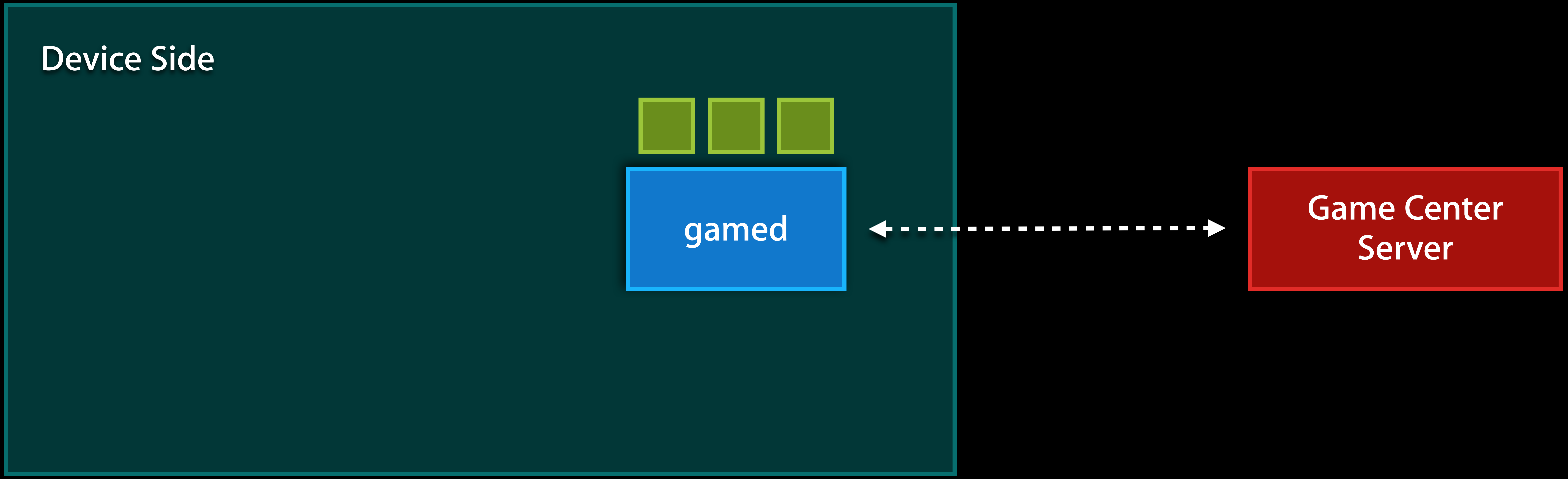
gamed

Game Center Server

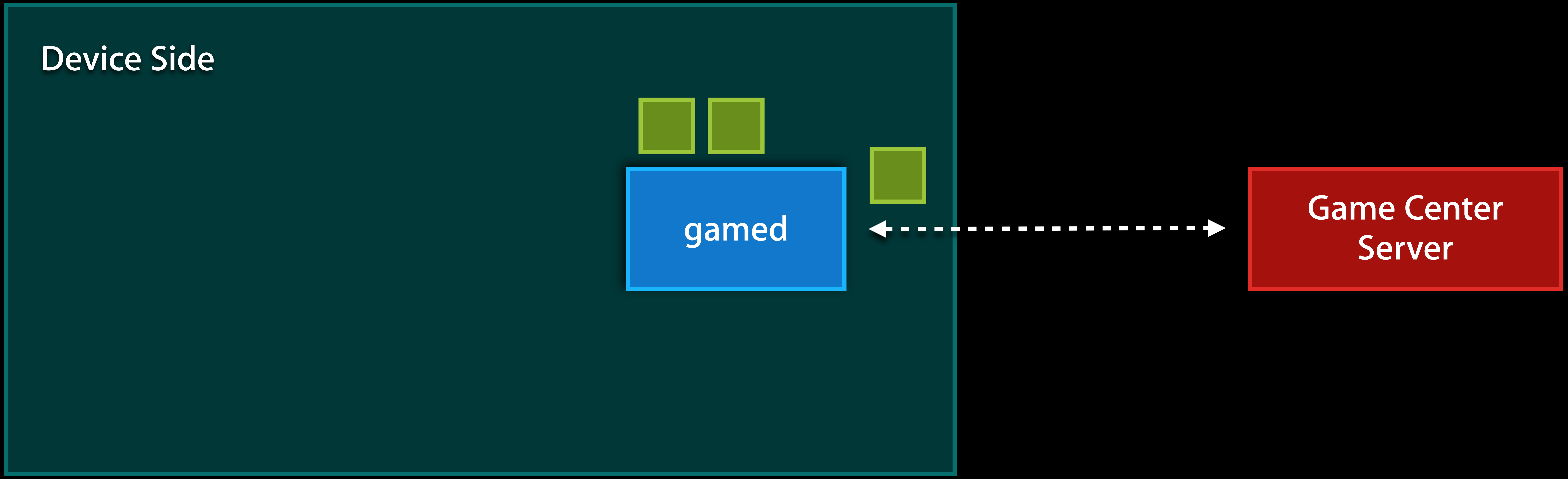
How Submission Works



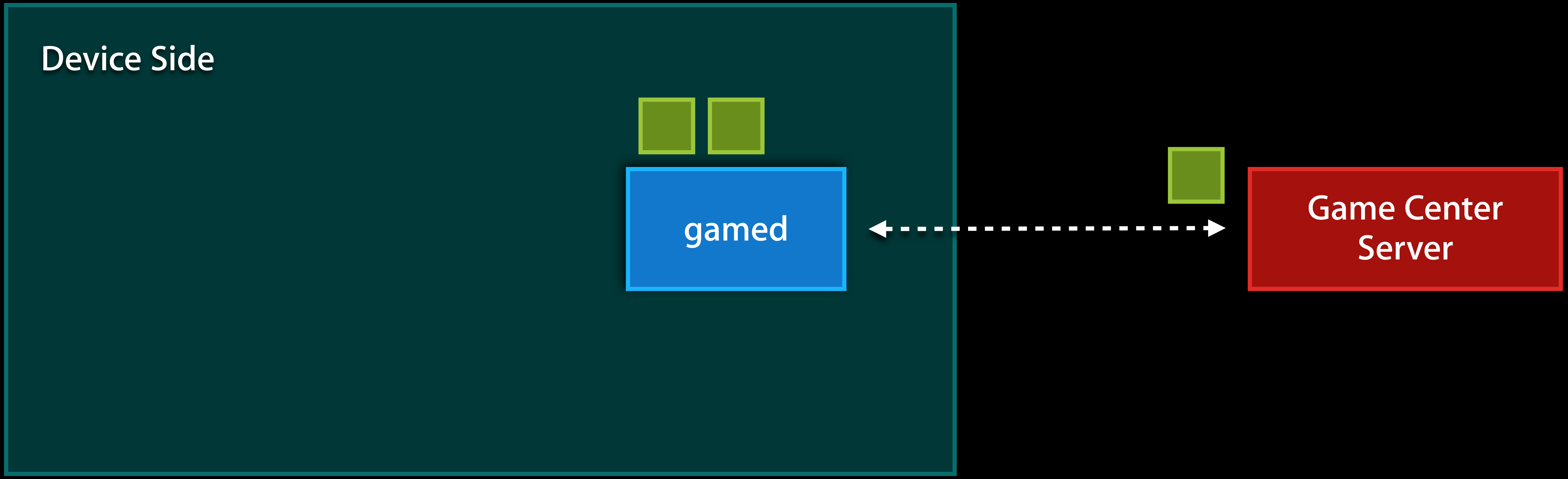
How Submission Works



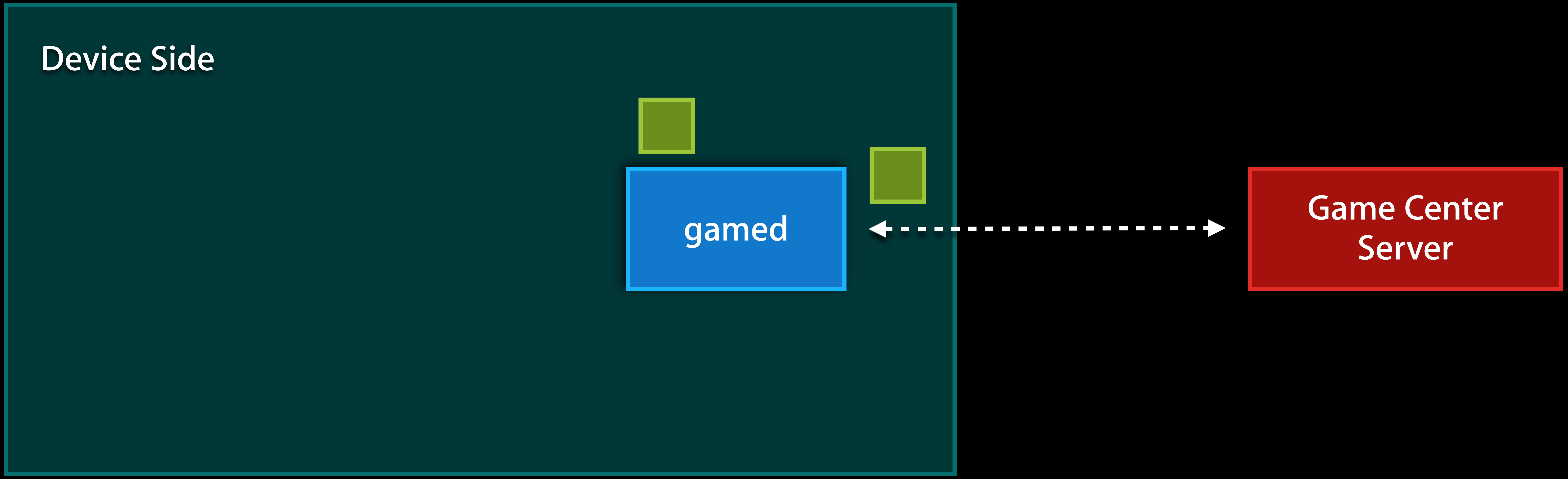
How Submission Works



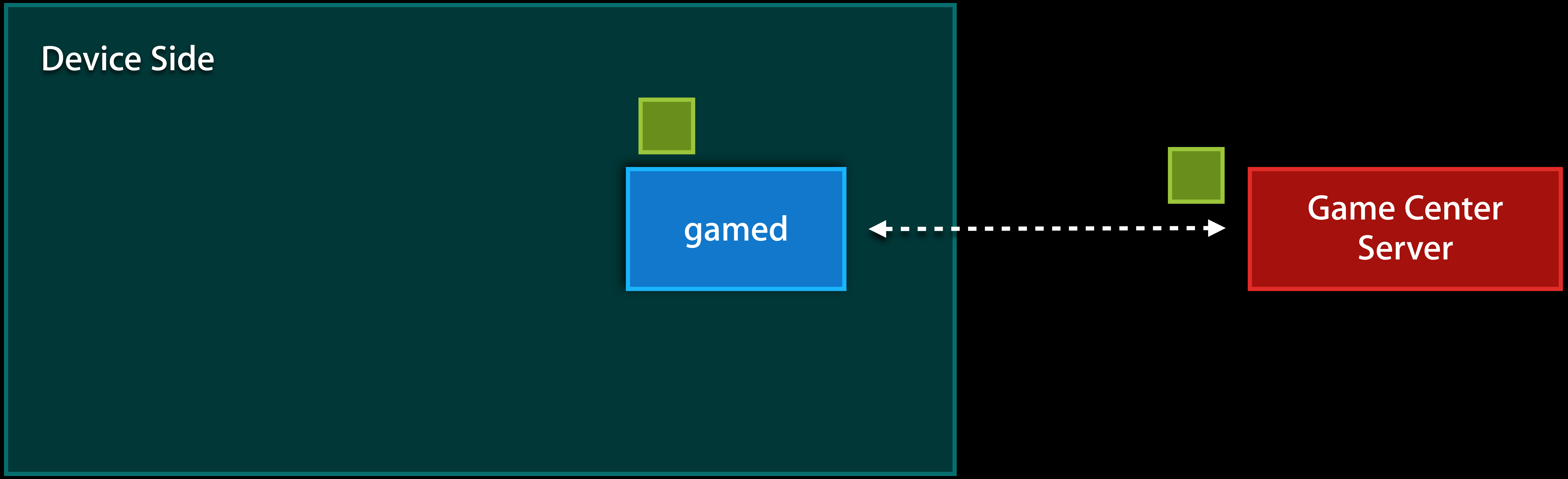
How Submission Works



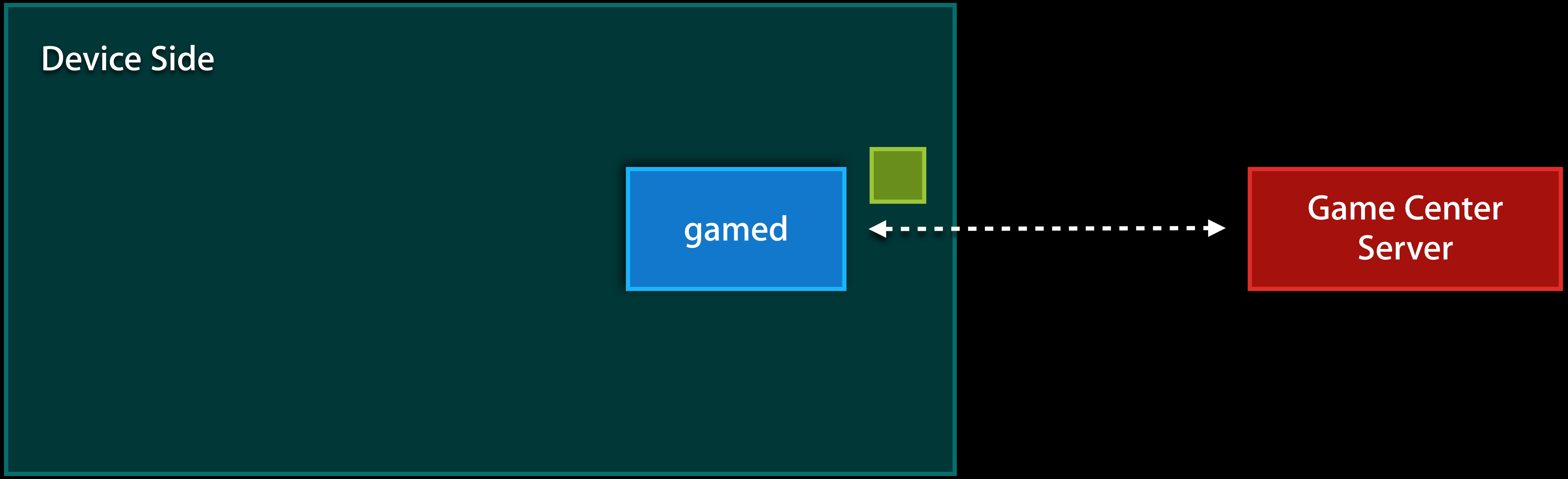
How Submission Works



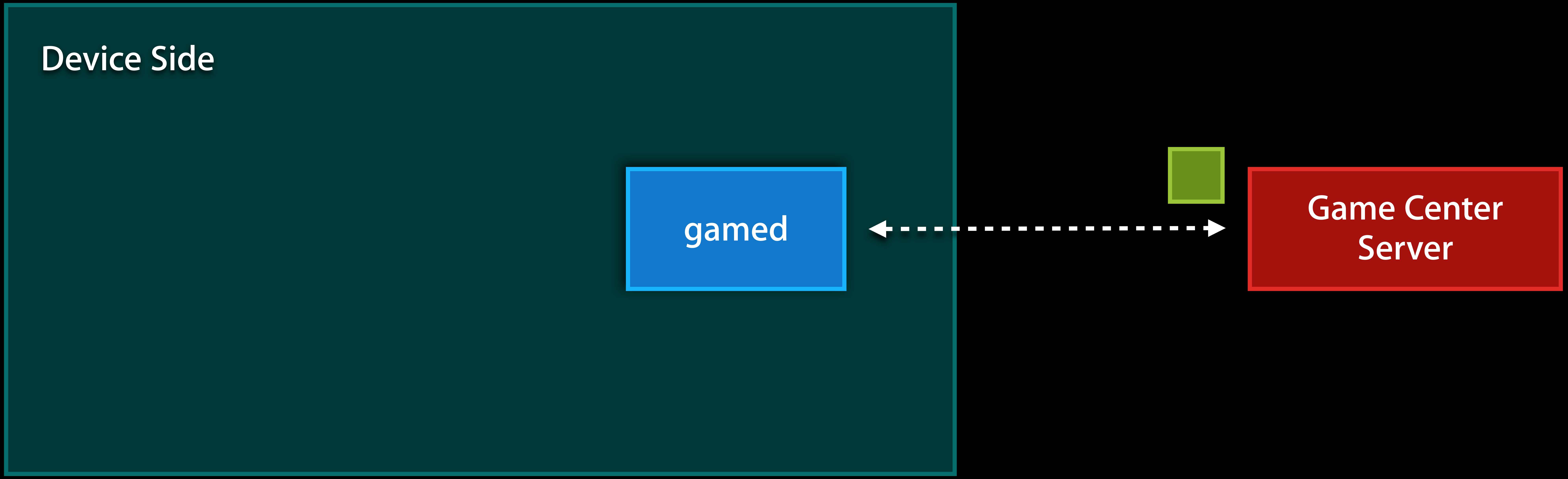
How Submission Works



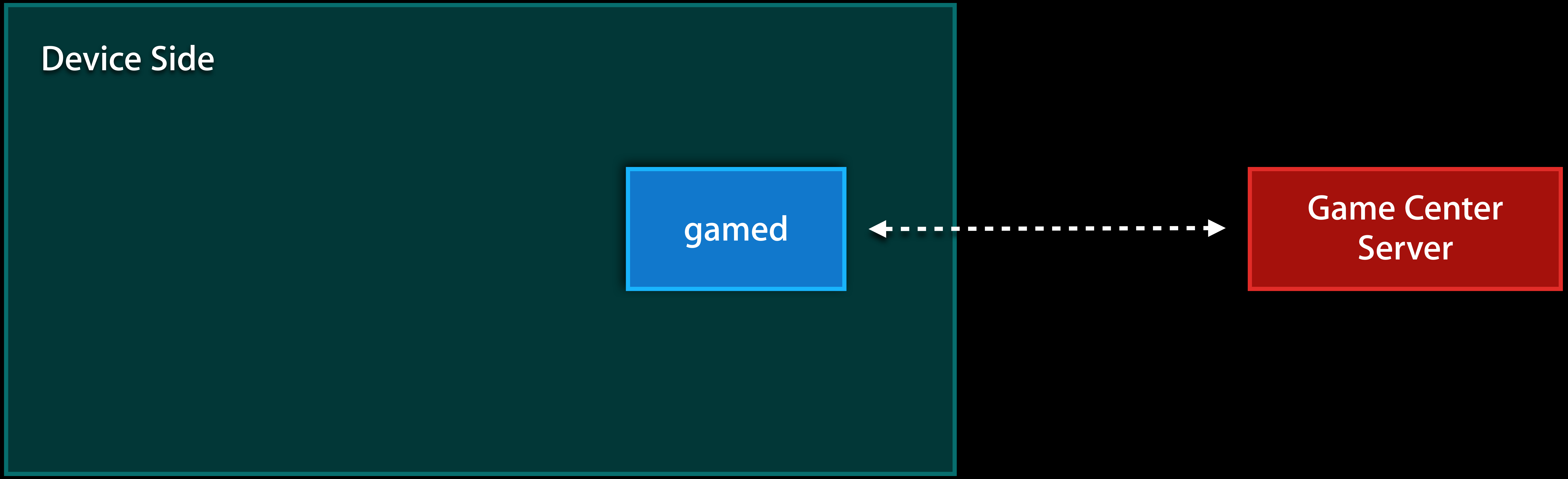
How Submission Works



How Submission Works



How Submission Works



How Submission Works

```
// Report
[GKScore reportScores: myScores withCompletionHandler: ^(NSError *error) {
    // ...
}];
```

How Submission Works

```
// Report
[GKScore reportScores: myScores withCompletionHandler: ^(NSError *error) {
    // ...
}];
```

How Submission Works

```
// Report
[GKScore reportScores: myScores withCompletionHandler: ^(NSError *error) {
    // ...
}];
```

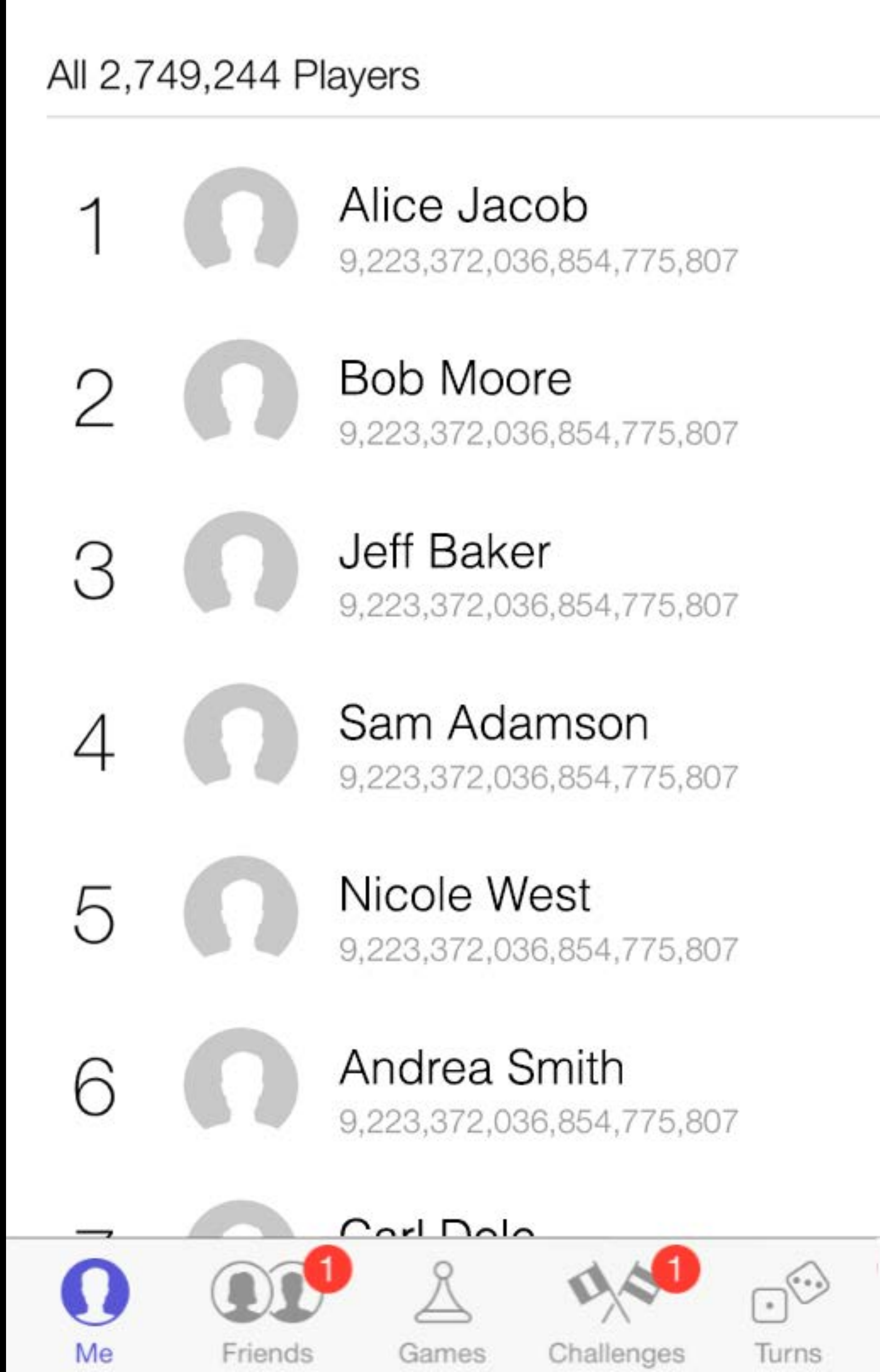
- Don’t cache scores and achievements
- Submit right away
- Batch your submissions

Leaderboard Recap

- More leaderboards
- Most Recent Score leaderboards
- Submit right away, no need to cache

Limiting Cheating

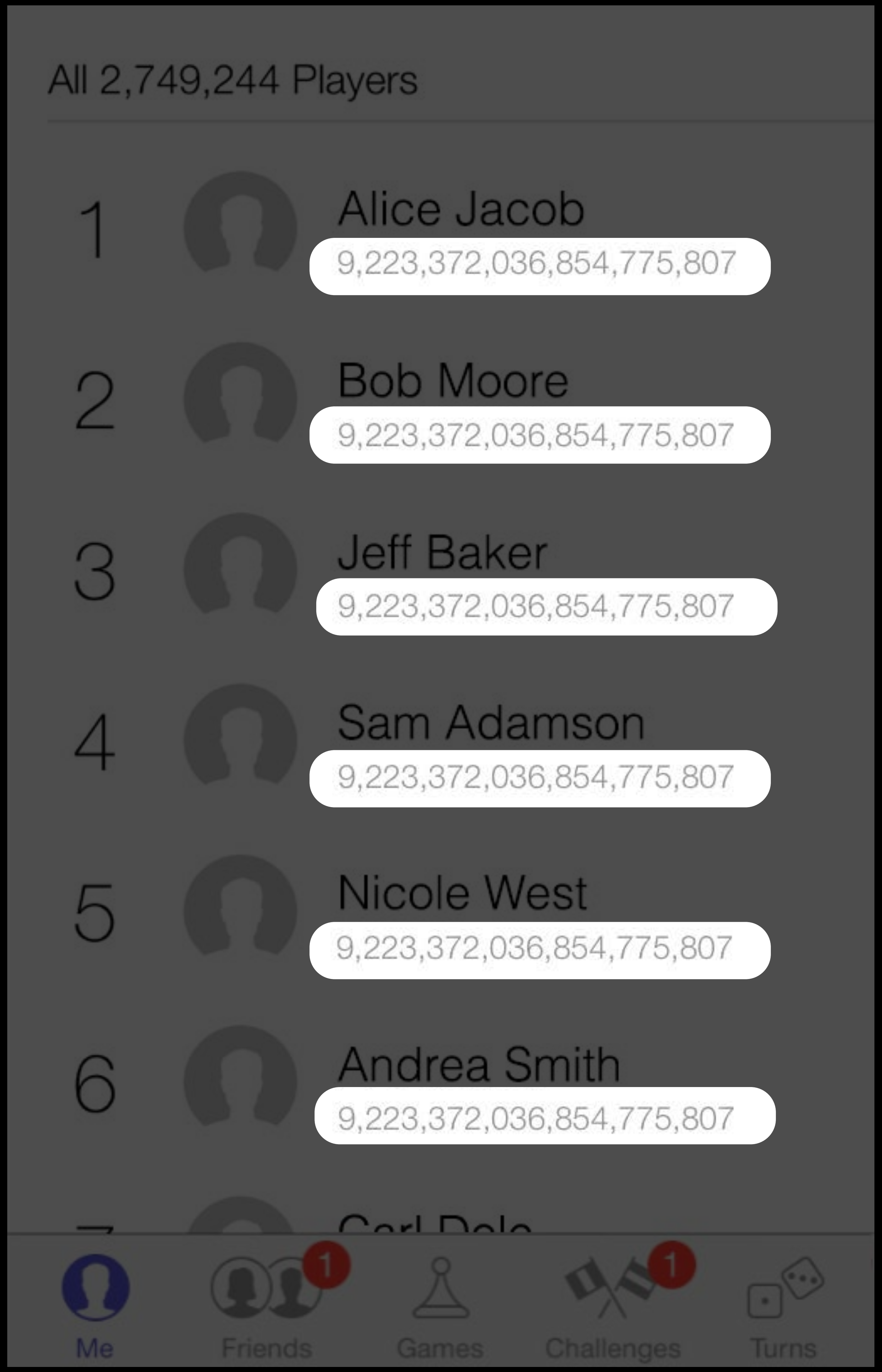
Limiting Cheating



All 2,749,244 Players

1	Alice Jacob 9,223,372,036,854,775,807
2	Bob Moore 9,223,372,036,854,775,807
3	Jeff Baker 9,223,372,036,854,775,807
4	Sam Adamson 9,223,372,036,854,775,807
5	Nicole West 9,223,372,036,854,775,807
6	Andrea Smith 9,223,372,036,854,775,807

Limiting Cheating

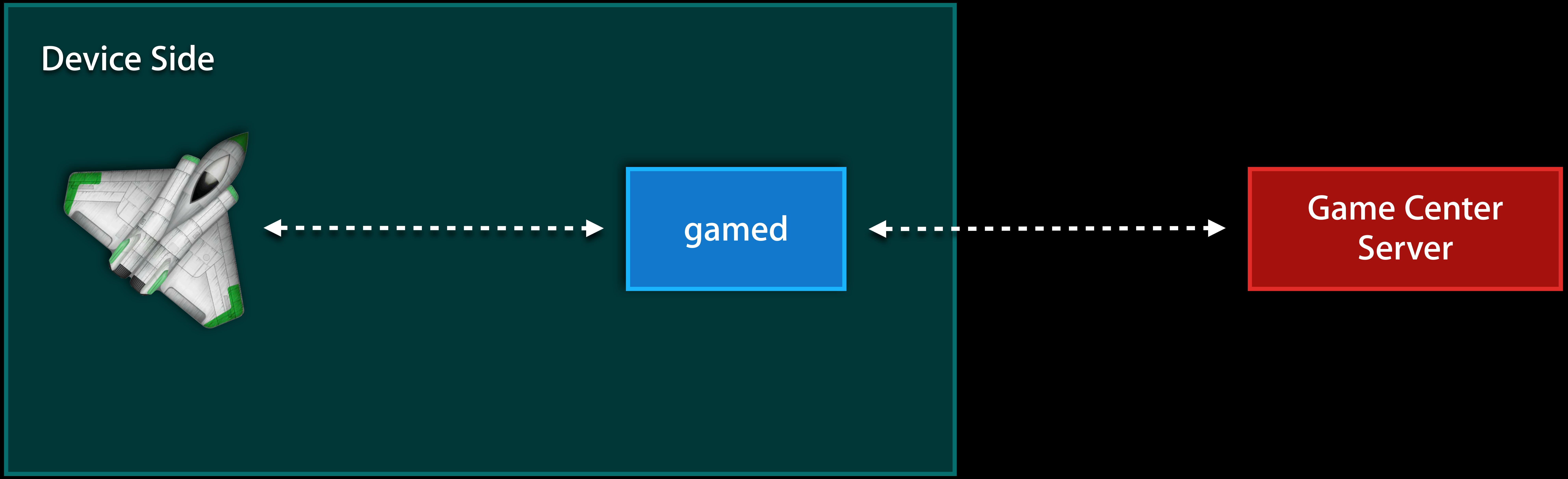


All 2,749,244 Players

1	Alice Jacob	9,223,372,036,854,775,807
2	Bob Moore	9,223,372,036,854,775,807
3	Jeff Baker	9,223,372,036,854,775,807
4	Sam Adamson	9,223,372,036,854,775,807
5	Nicole West	9,223,372,036,854,775,807
6	Andrea Smith	9,223,372,036,854,775,807
7	Carl Dale	

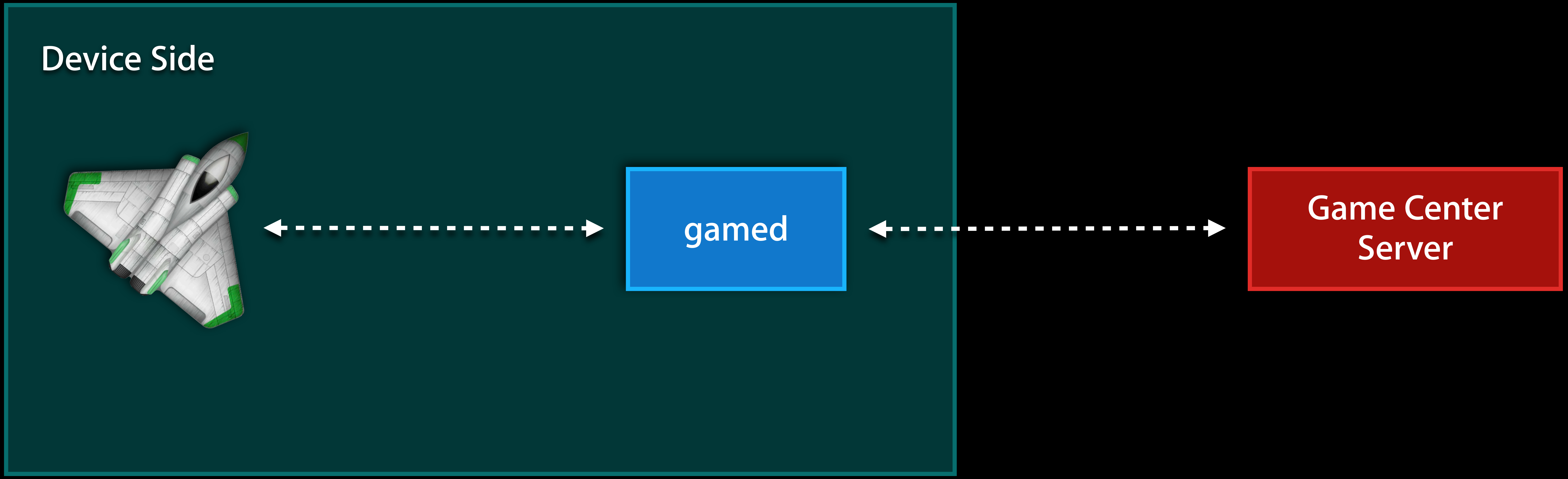
Free for You: Signed Submissions

Limiting Cheating



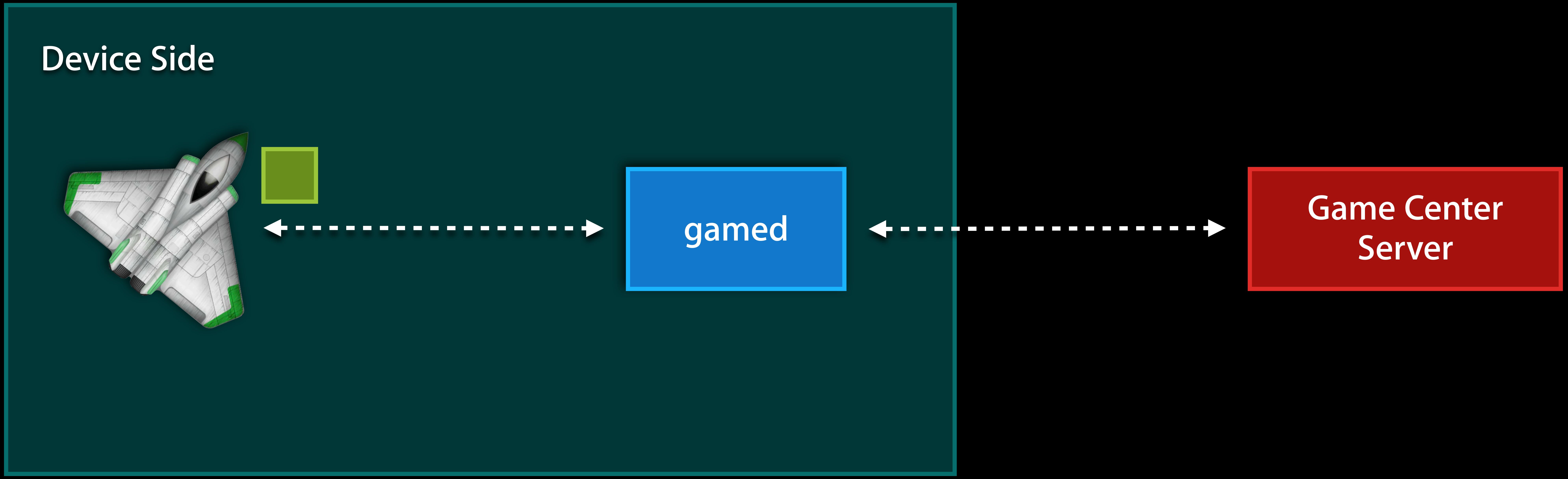
Free for You: Signed Submissions

Limiting Cheating



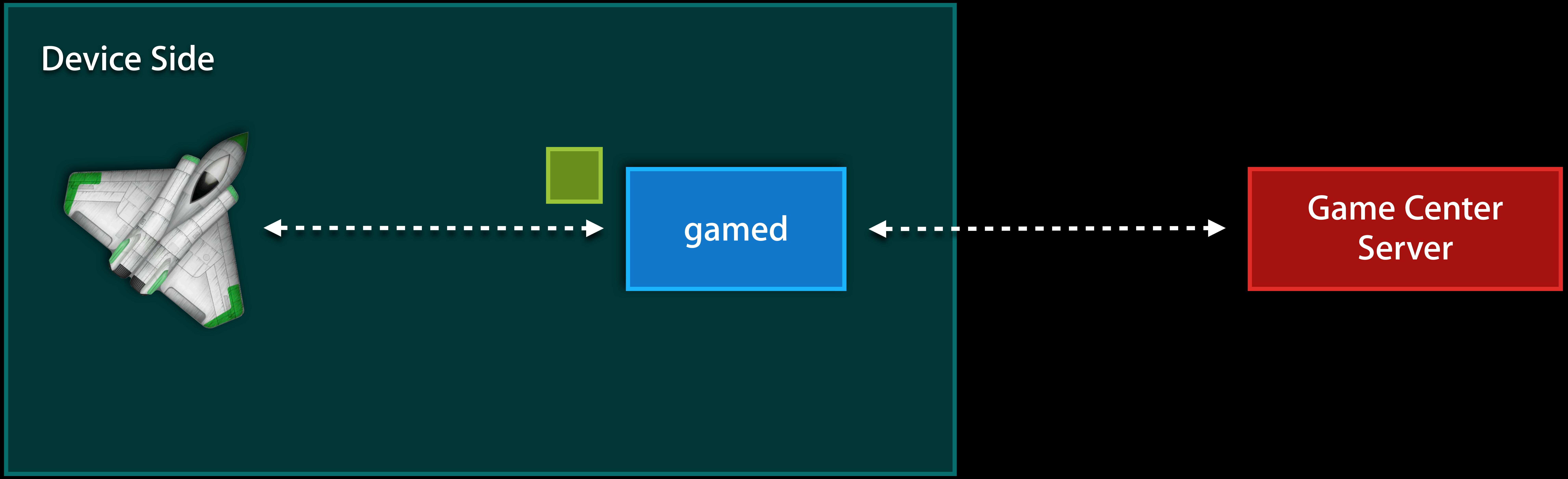
Free for You: Signed Submissions

Limiting Cheating



Free for You: Signed Submissions

Limiting Cheating



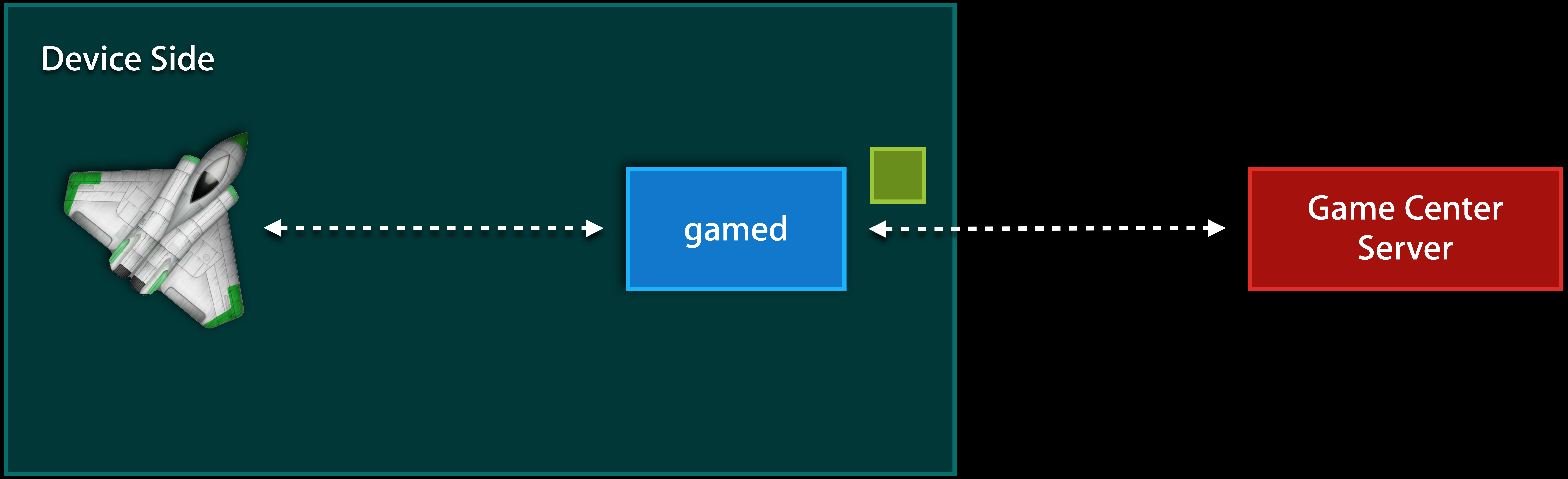
Device Side

gamed

Game Center Server

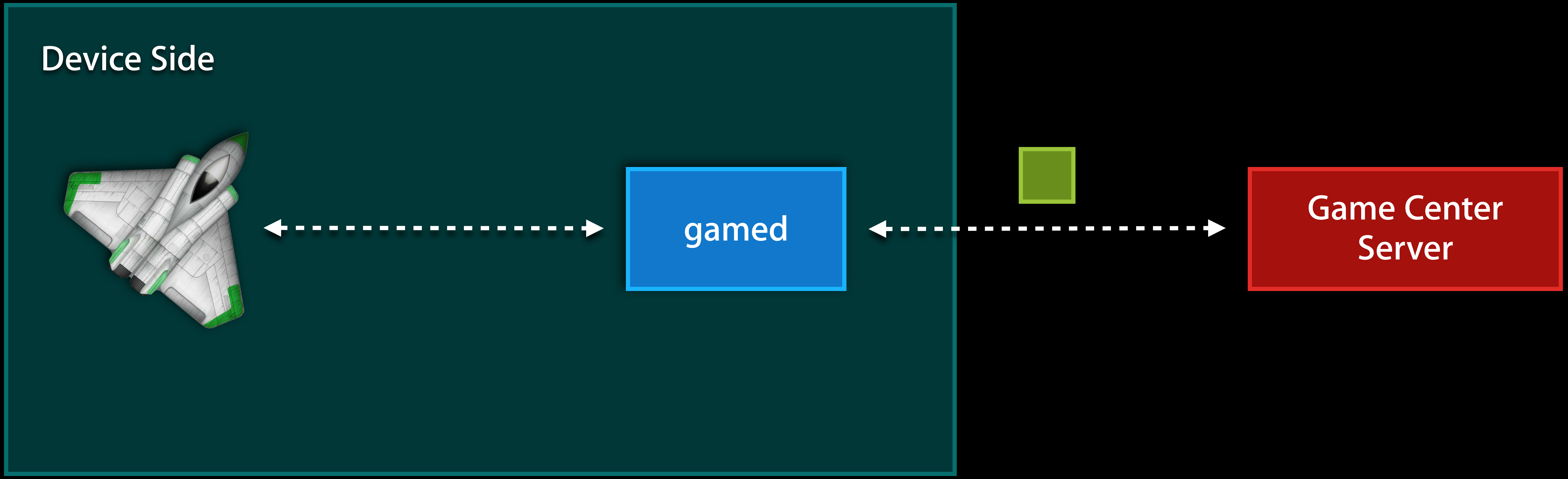
Free for You: Signed Submissions

Limiting Cheating



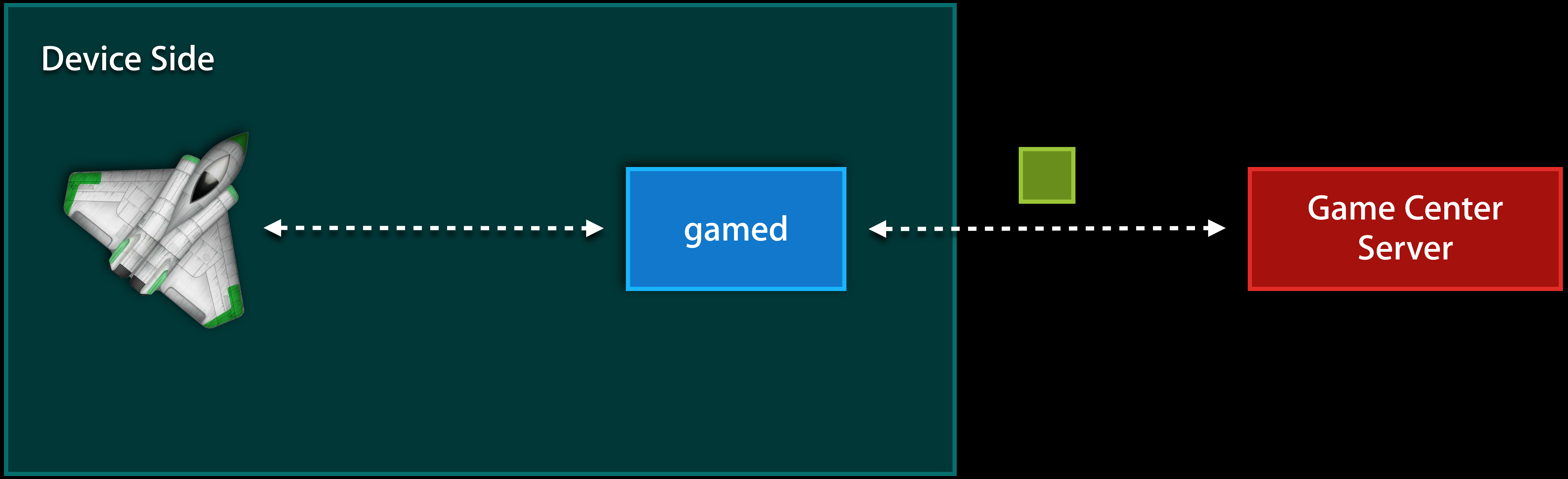
Free for You: Signed Submissions

Limiting Cheating



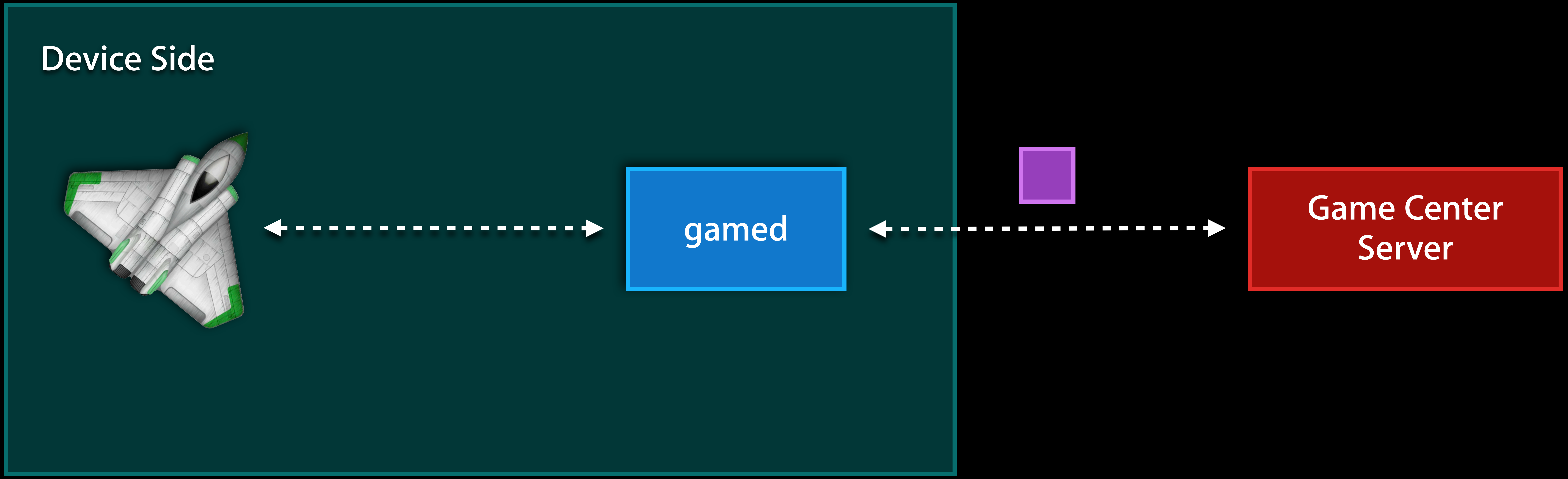
Free for You: Signed Submissions

Limiting Cheating



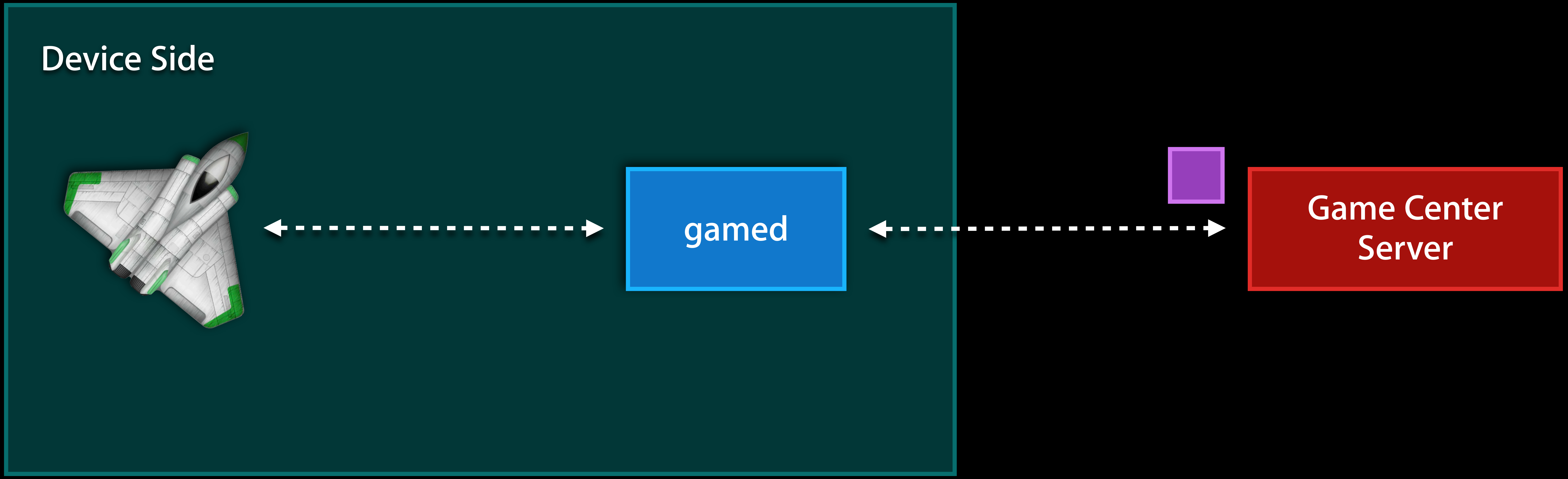
Free for You: Signed Submissions

Limiting Cheating



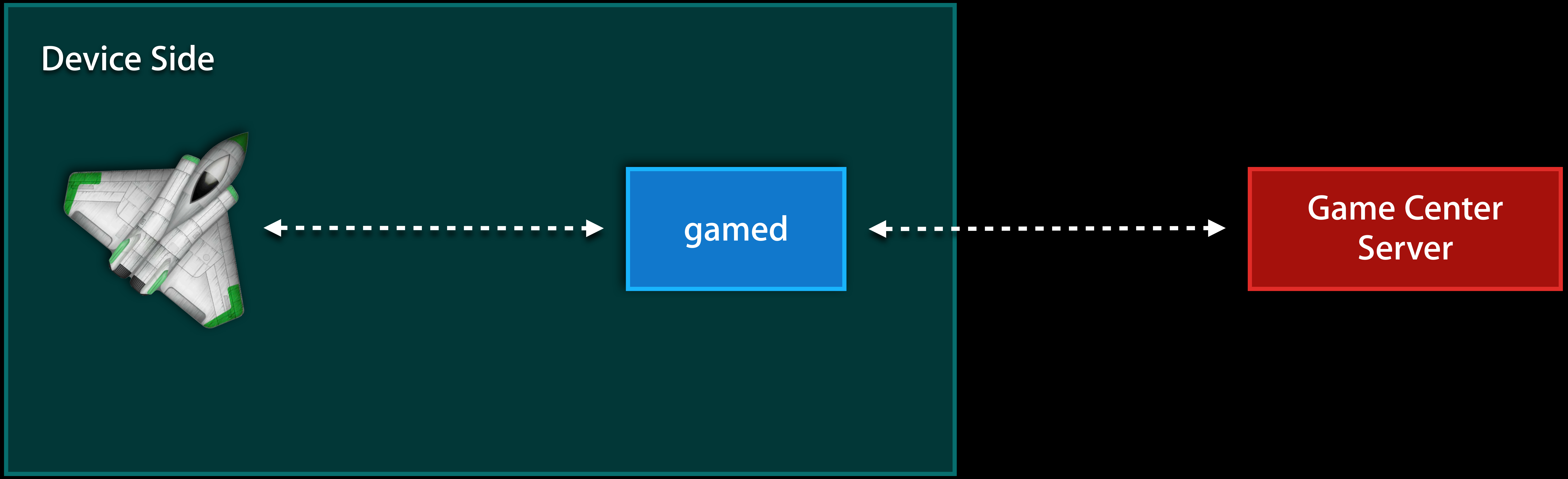
Free for You: Signed Submissions

Limiting Cheating



Free for You: Signed Submissions

Limiting Cheating



Max Scores

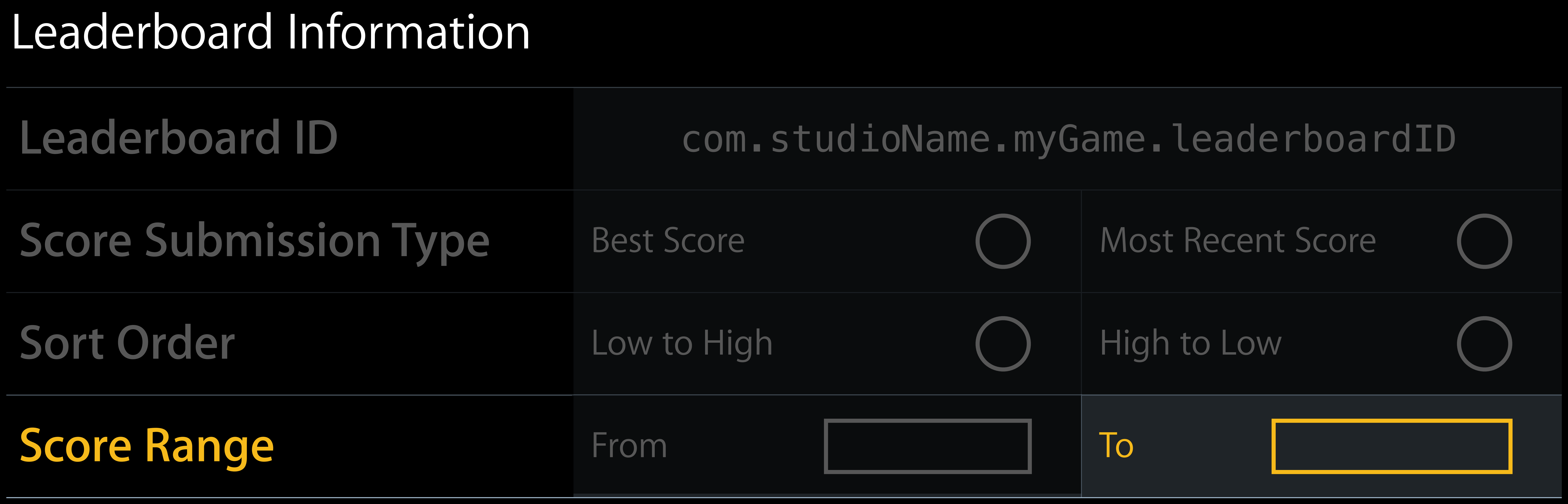
Limiting Cheating

Leaderboard Information

Leaderboard ID	`com.studioName.myGame.leaderboardID`	
Score Submission Type	Best Score ○	Most Recent Score ○
Sort Order	Low to High ○	High to Low ○
Score Range	From □	To □

Max Scores

Limiting Cheating



Leaderboard Information

Leaderboard ID	com.studioName.myGame.leaderboardID			
Score Submission Type	Best Score	○	Most Recent Score	○
Sort Order	Low to High	○	High to Low	○
Score Range	From	□	To	□

Player Management

Limiting Cheating

- New iTunes Connect tool
- Actively manage and audit suspicious activity
- What’s New in iTunes Connect

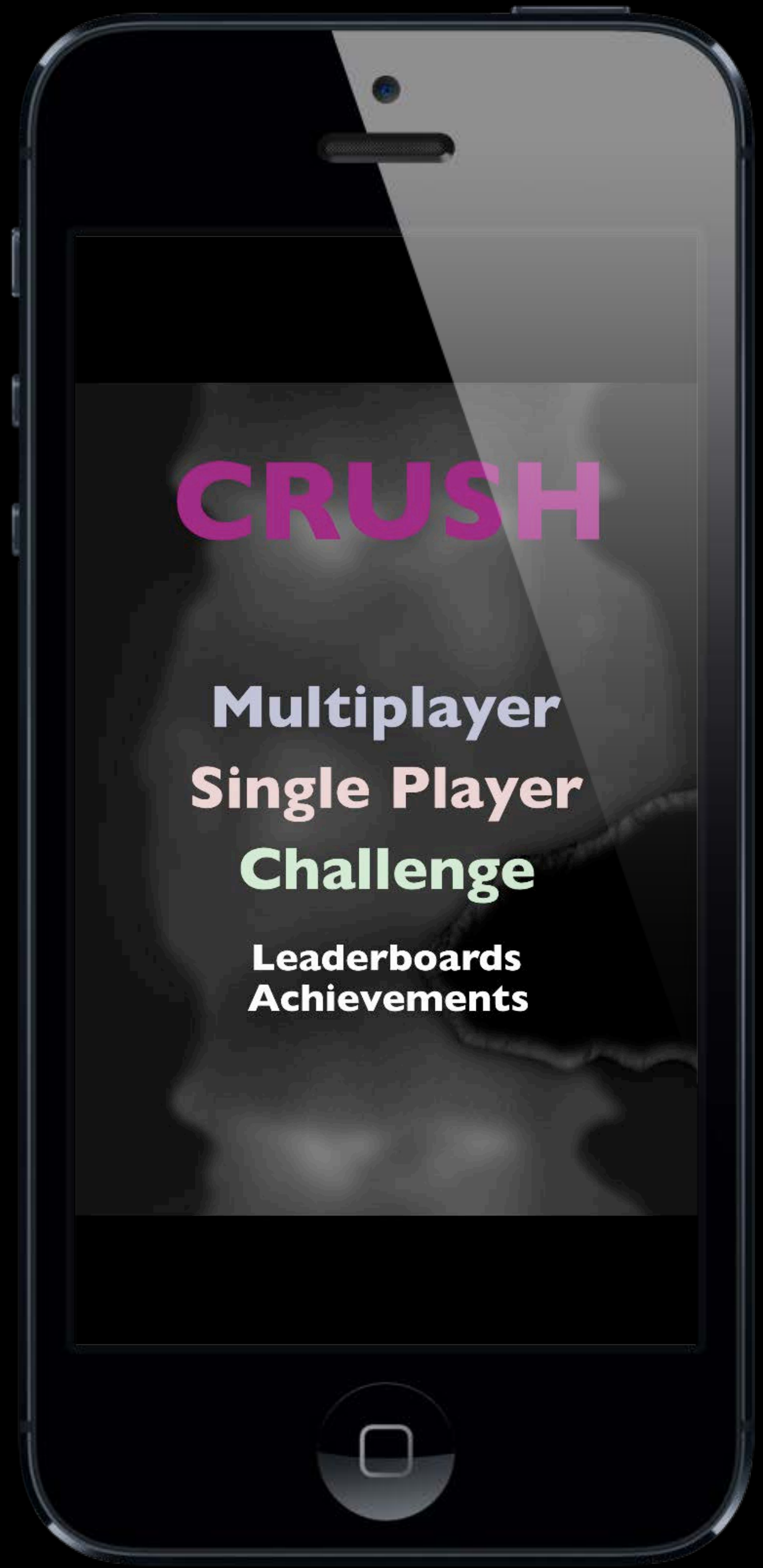
Security Recap

- Signed submissions
- Use Score Range to filter out cheaters
- New management tool

Challenges

Challenges Recap

- Introduced with iOS 6 and Mountain Lion
- Adds competitive element for any game
- Supported automatically with built-in UI
    - API available for extension



CRUSH

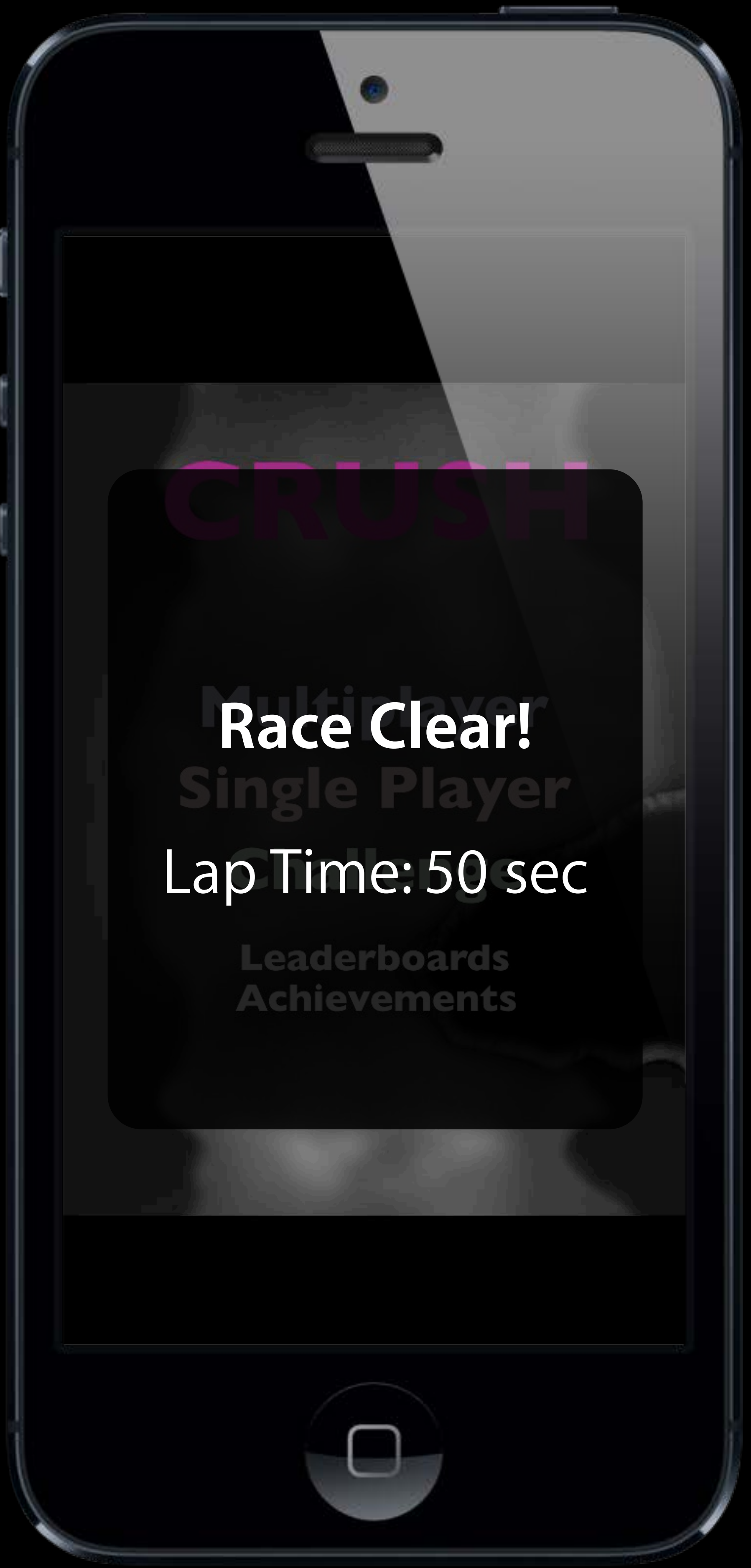
Multiplayer
Single Player
Challenge

Leaderboards
Achievements

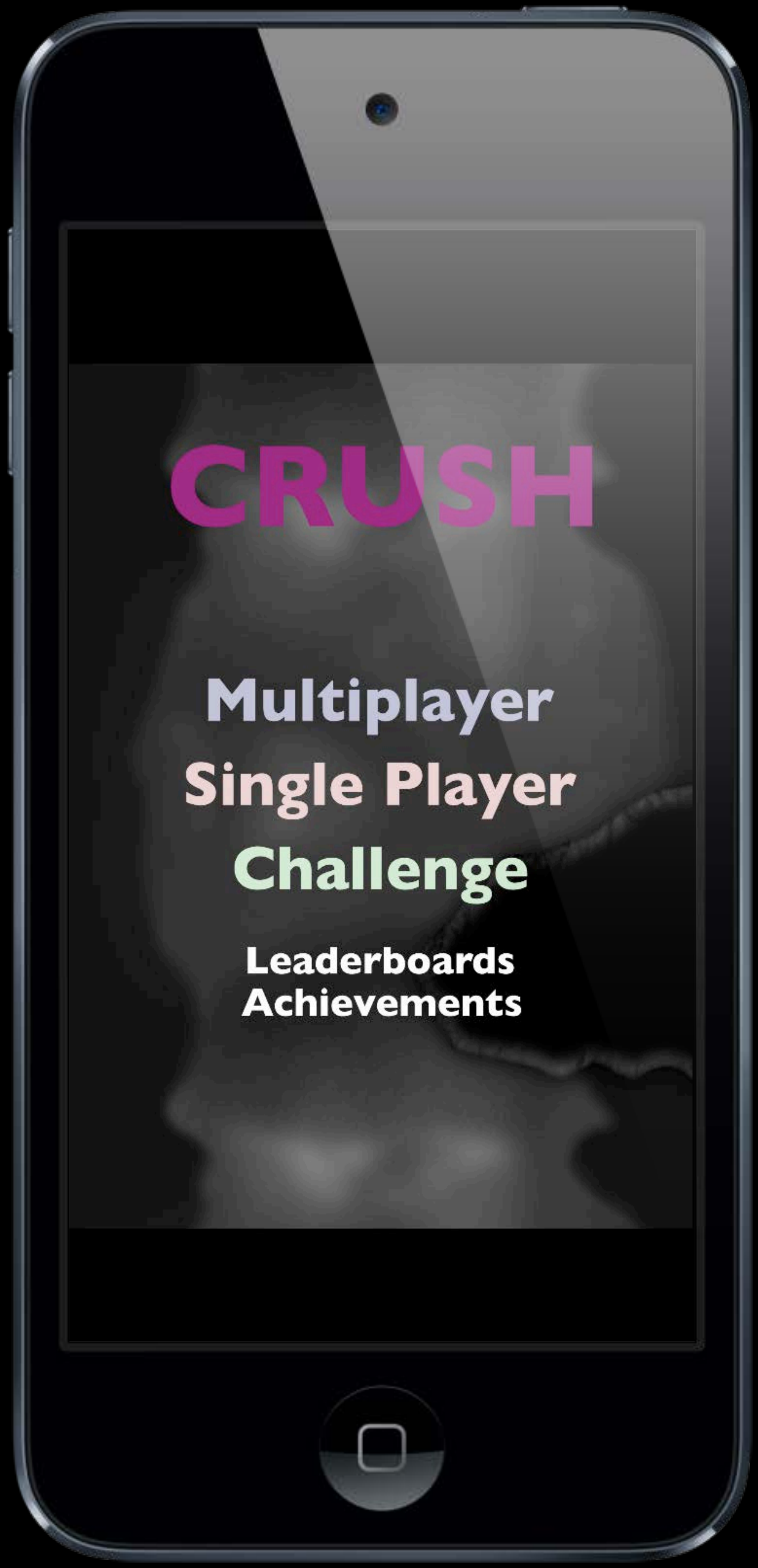
CRUSH

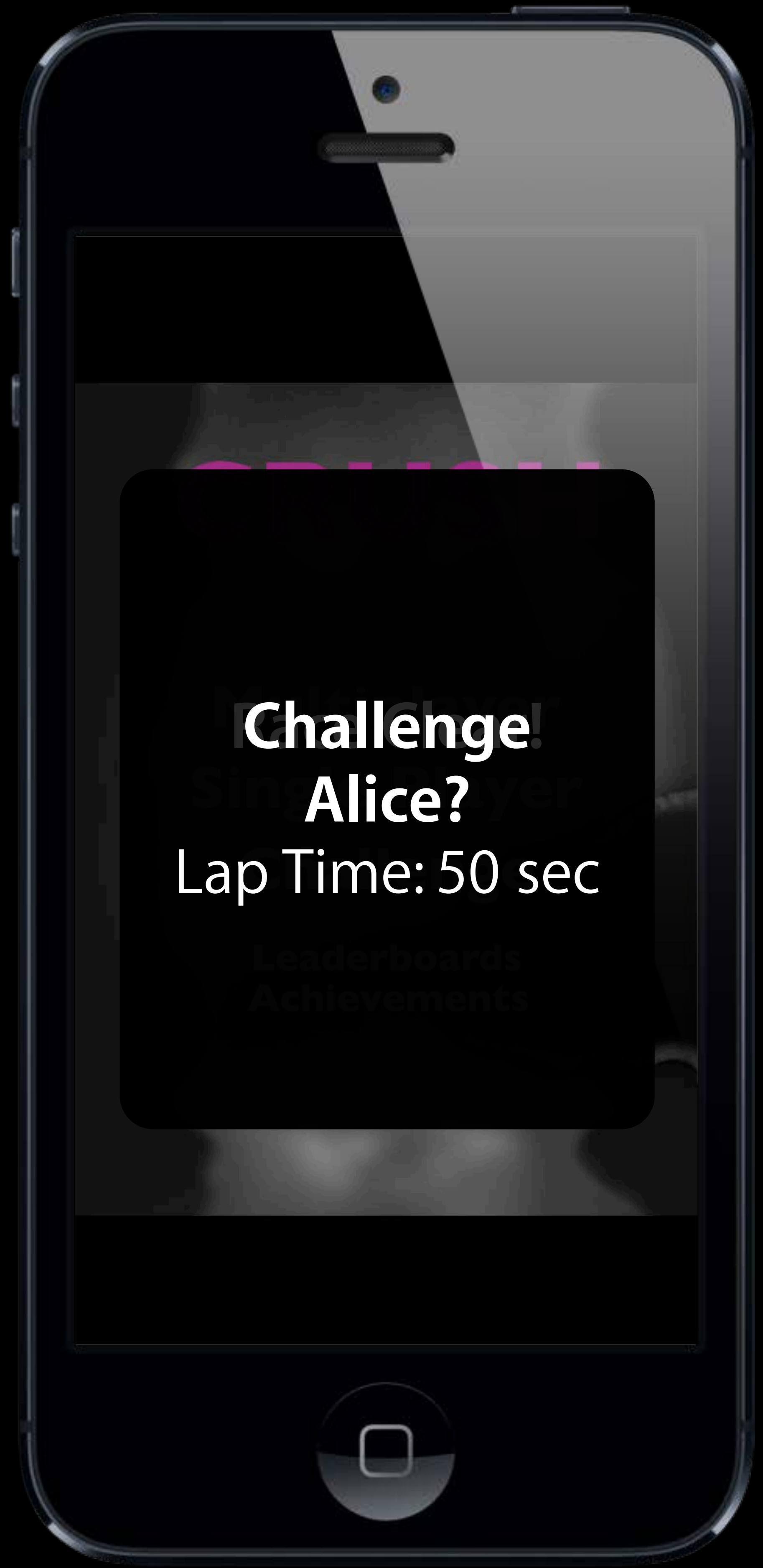
Multiplayer
Single Player
Challenge

Leaderboards
Achievements

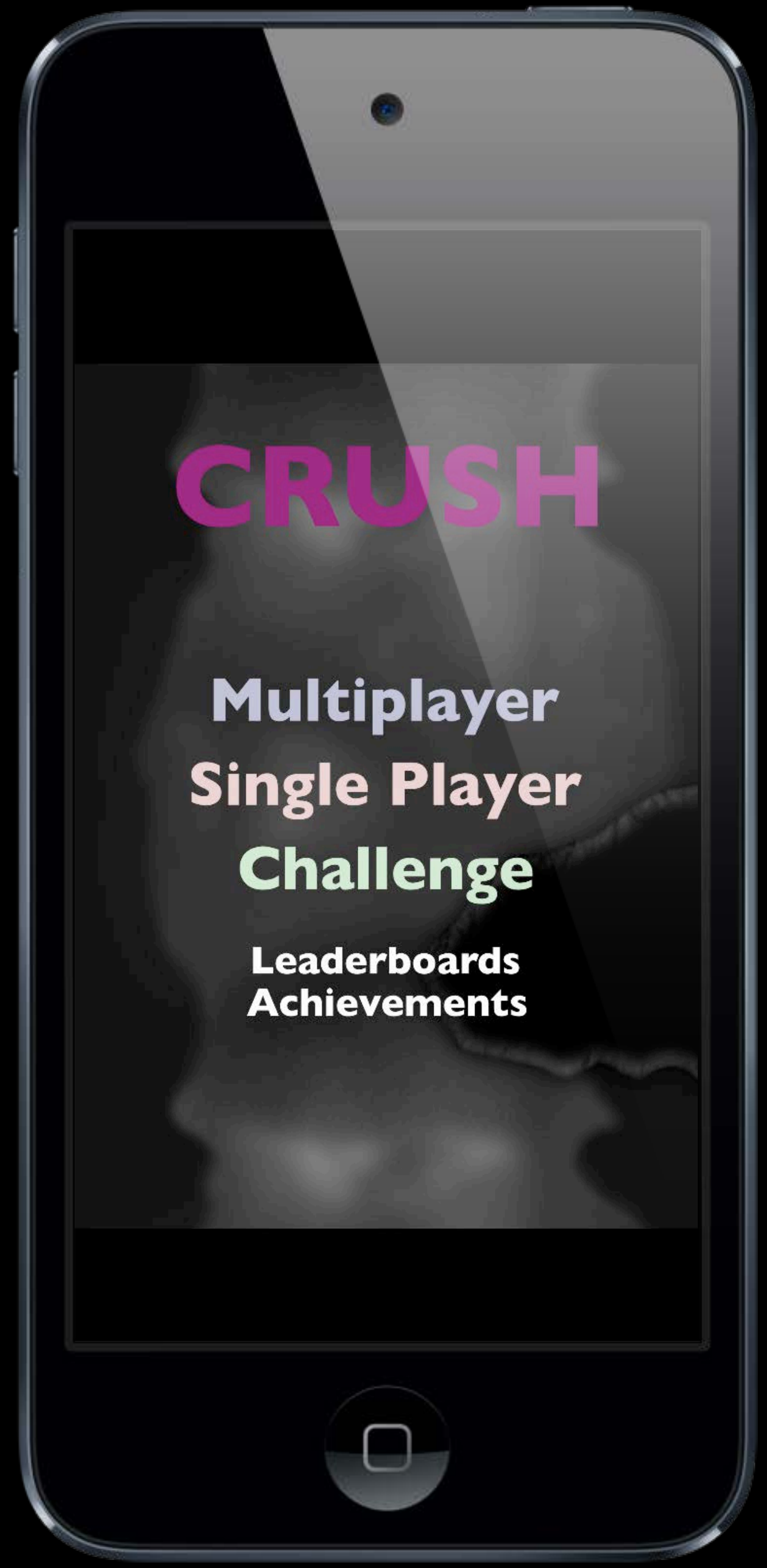

Race Clear!

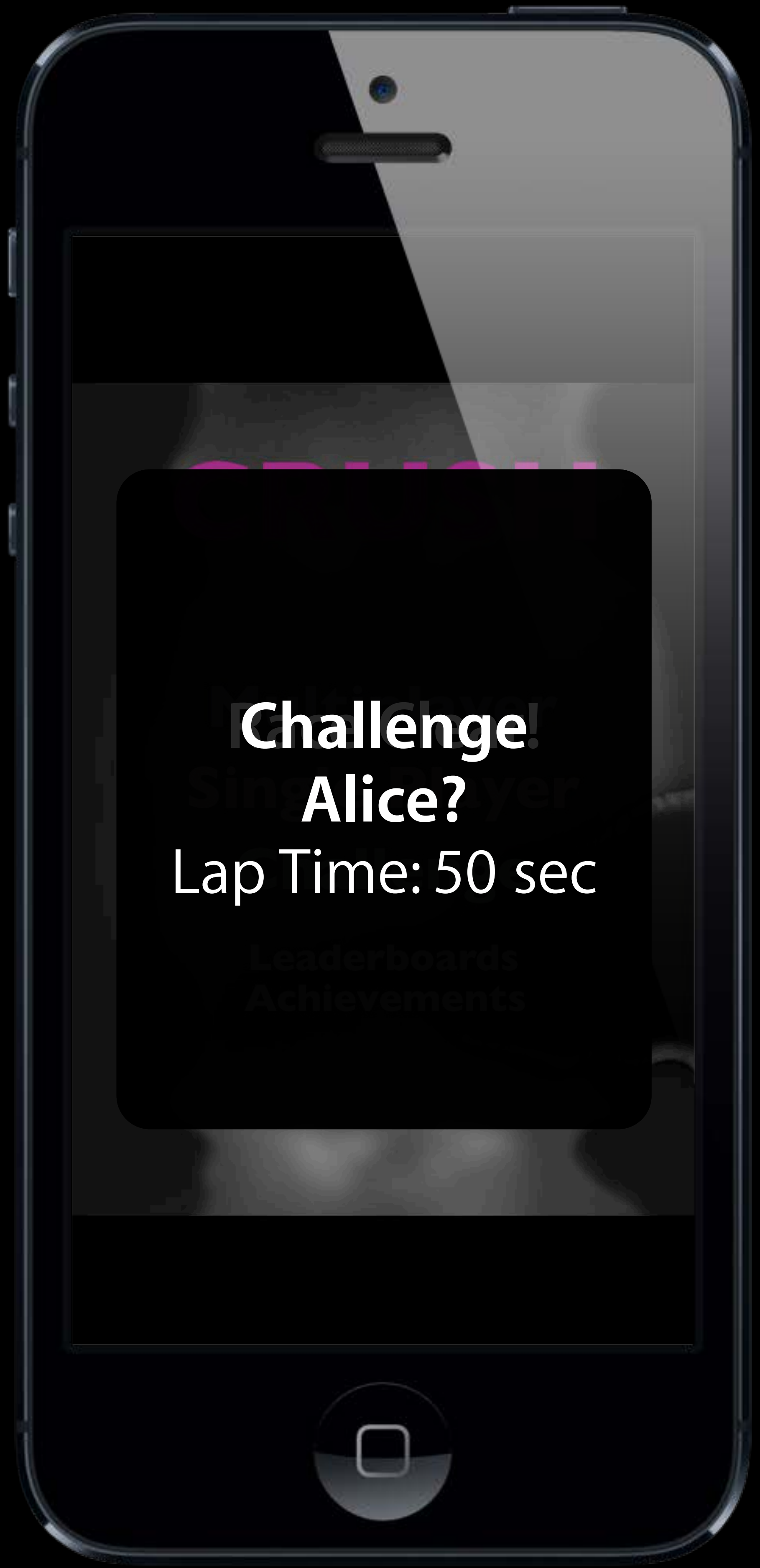
Lap Time: 50 sec



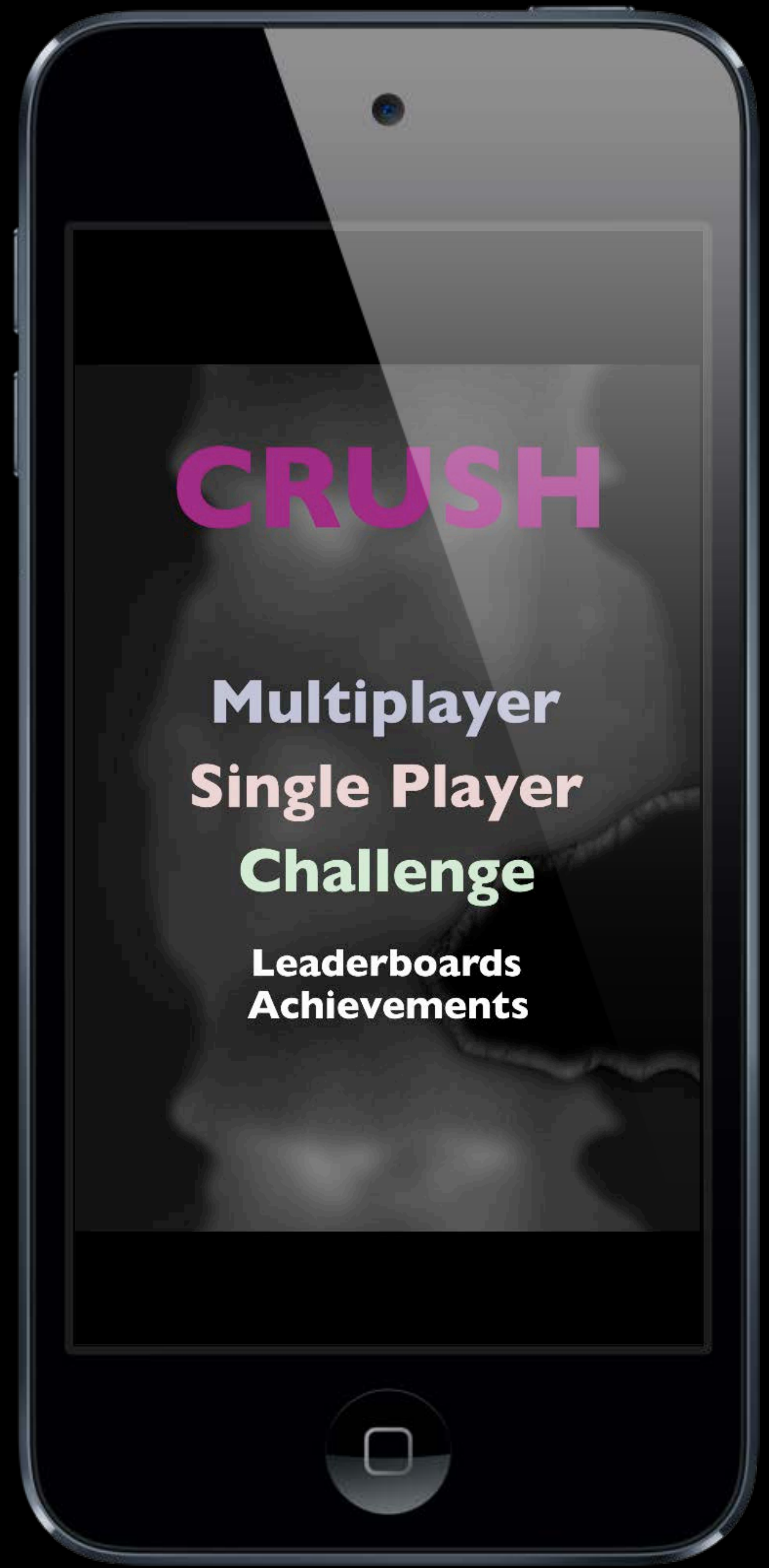


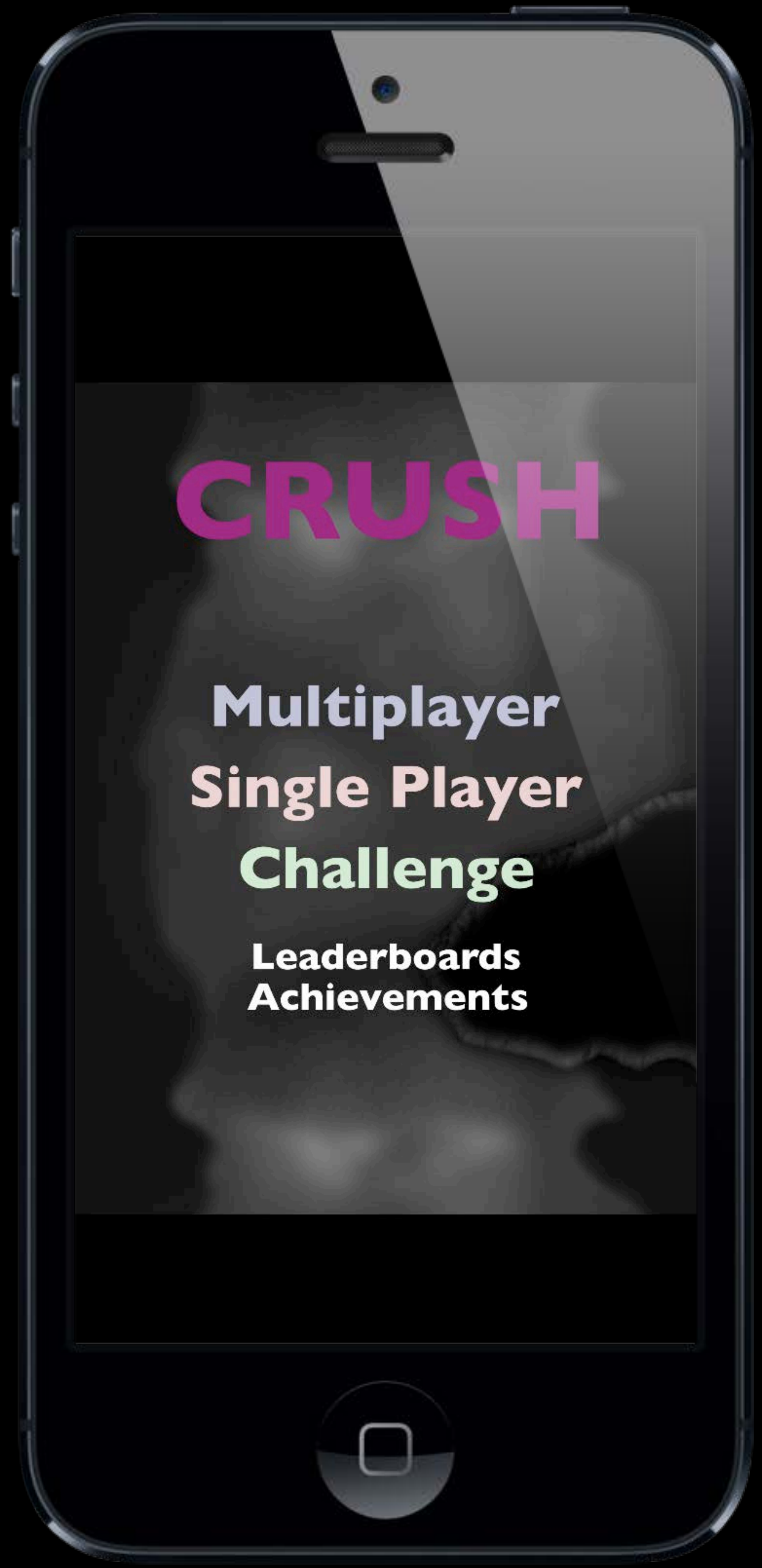
Challenge Alice?
Lap Time: 50 sec



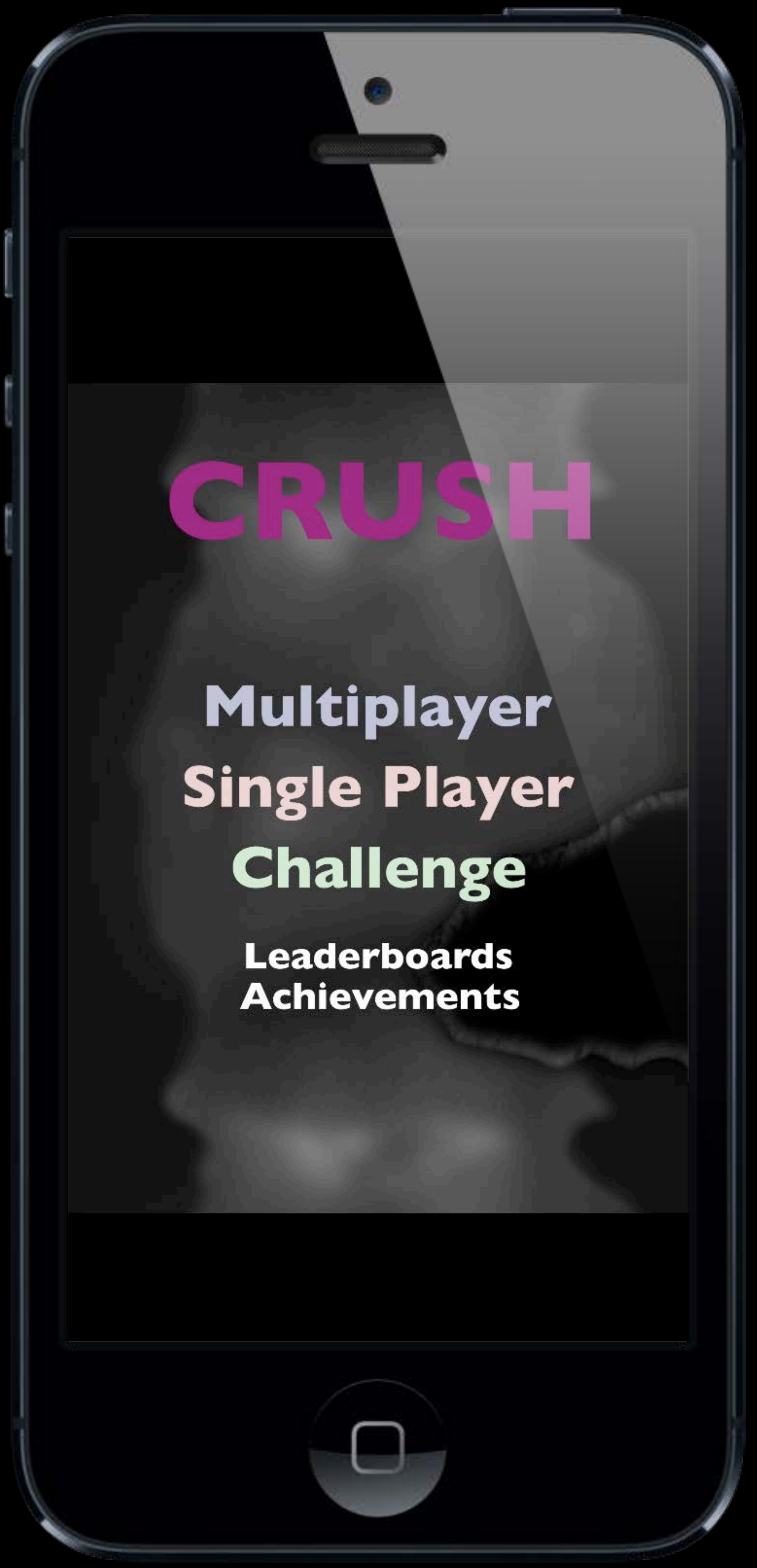


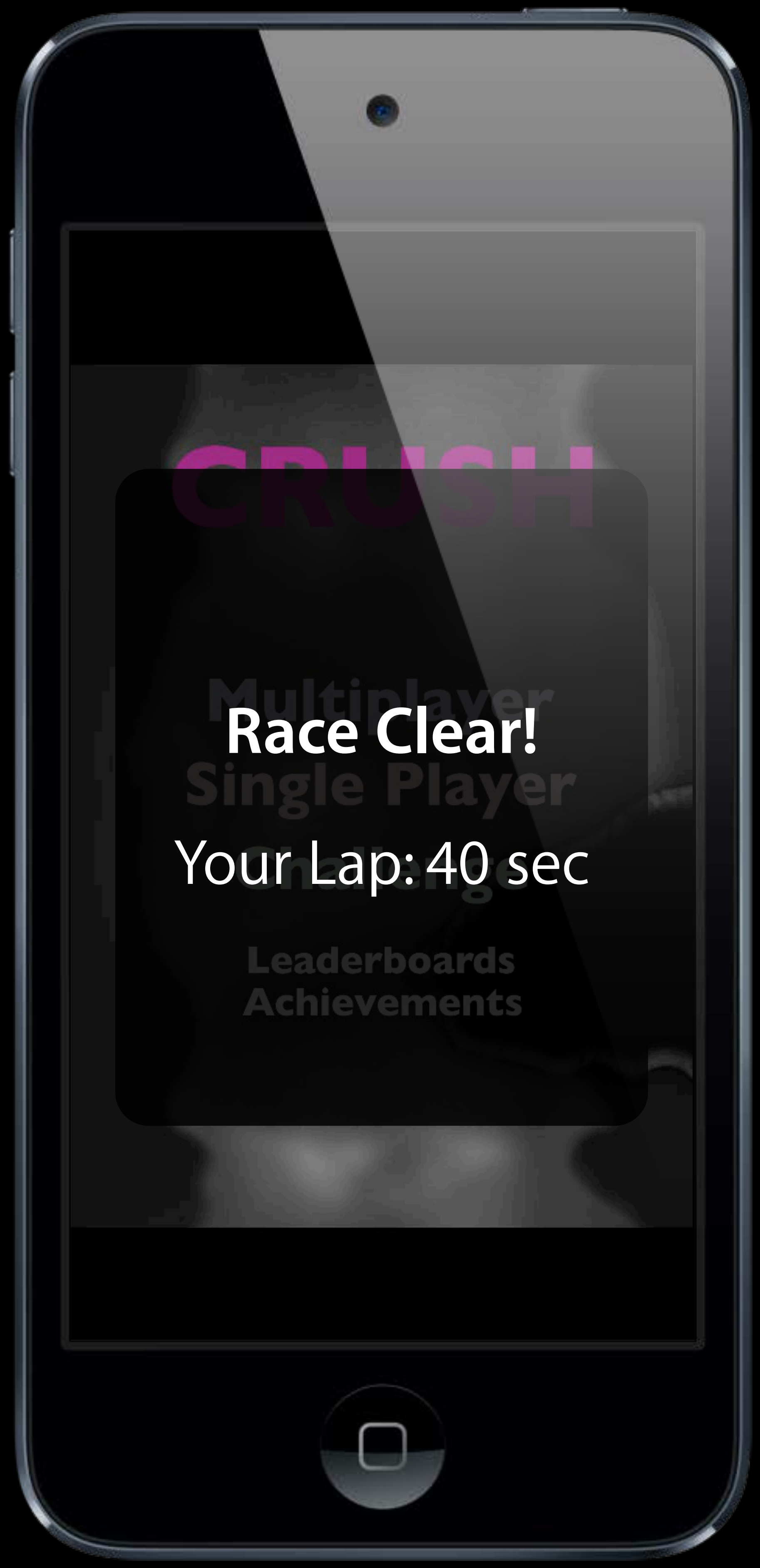
Challenge Alice?
Lap Time: 50 sec




CRUSH
Multiplayer
Single Player
Challenge
Leaderboards
Achievements

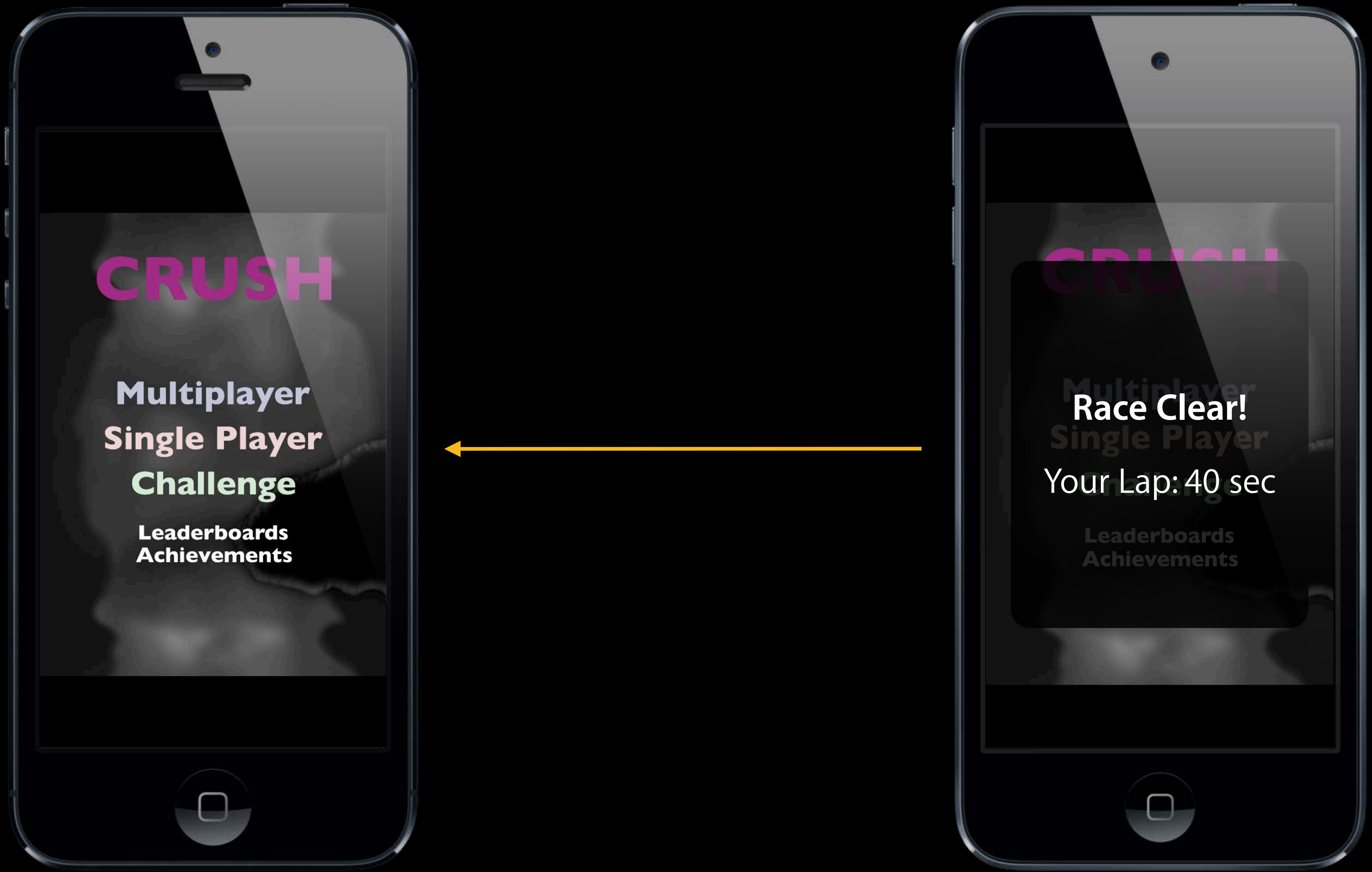
CRUSH
Multiplayer
Single Player
Challenge
Leaderboards
Achievements





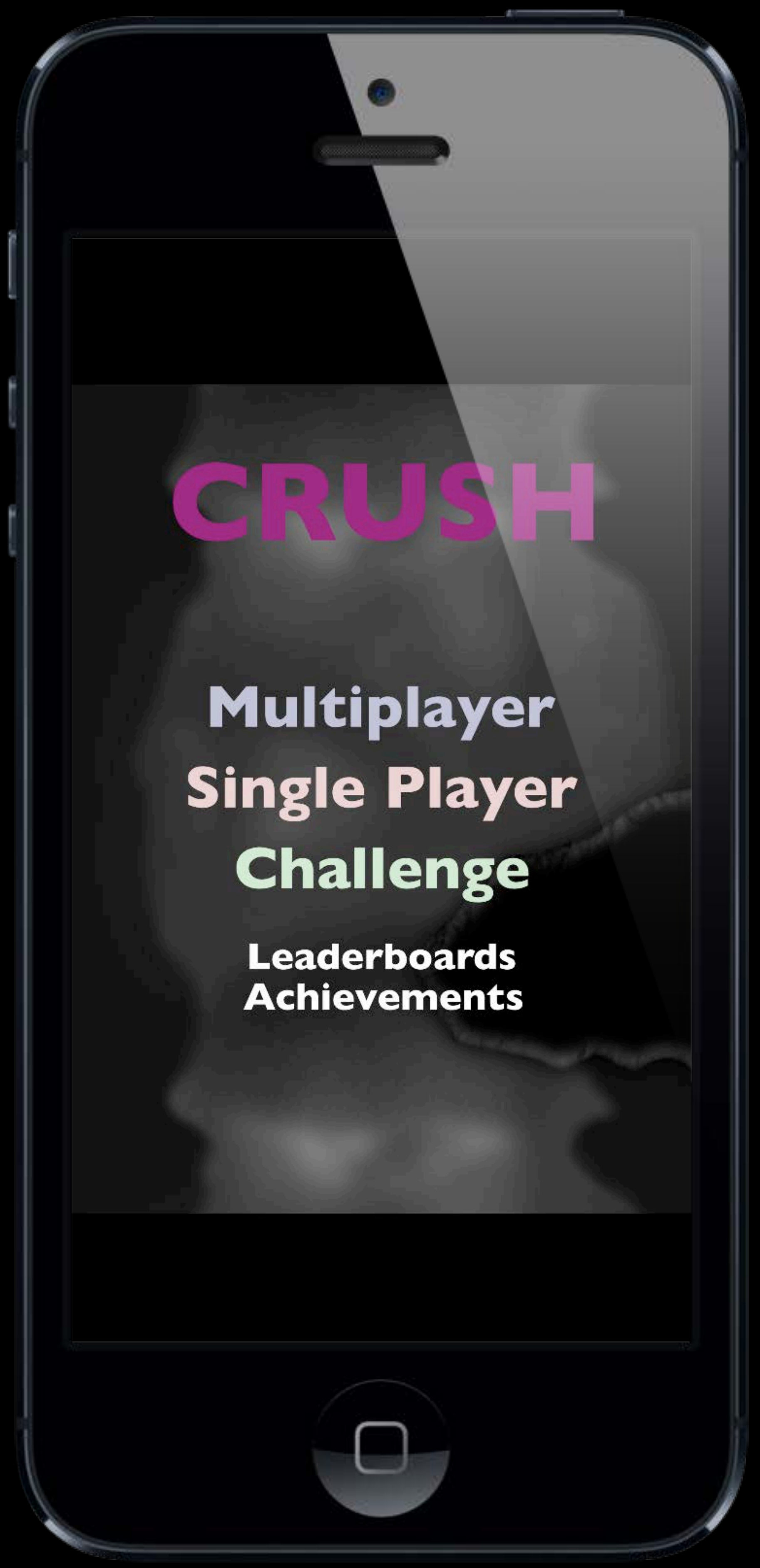
Race Clear!

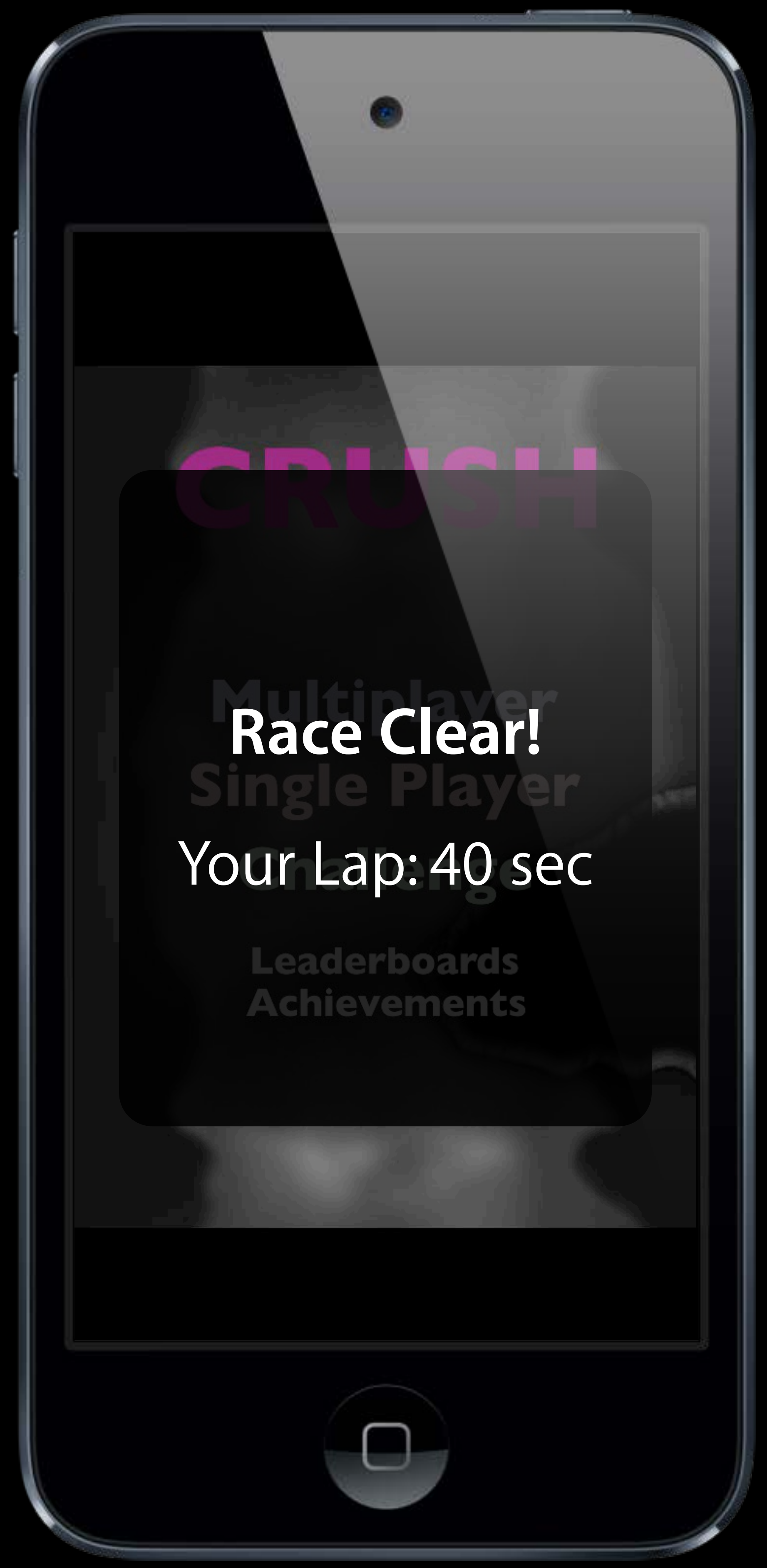
Your Lap: 40 sec



Race Clear!

Your Lap: 40 sec




Race Clear!

Your Lap: 40 sec

Completing Challenges

Modal Completion

- Selectively complete specific challenges
- New API for submission

```
reportScores:withEligibleChallenges:withCompletionHandler:
reportAchievements:withEligibleChallenges:withCompletionHandler:
```

Modal Completion

Score Challenge example

```
// load challenges for current player
[GKChallenge loadReceivedChallengesWithCompletionHandler:
^(NSArray *challenges, NSError *error) {

    NSArray *scoreChallenges = [self filterForScoreChallenges: challenges];

    for (GKScoreChallenge *scoreChallenge in scoreChallenges) {
        // get car challenge was created with
        uint64_t context = scoreChallenge.score.context;
        CarModel carModel = [self carModelForContext: context];

        if (carModel == [self carModel]) {
            [challengesWithSameCar addObject: scoreChallenge];
        }
    }
    // continued...
```

Modal Completion

Score Challenge example

```
// load challenges for current player
[GKChallenge loadReceivedChallengesWithCompletionHandler:
^(NSArray *challenges, NSError *error) {

    NSArray *scoreChallenges = [self filterForScoreChallenges: challenges];

    for (GKScoreChallenge *scoreChallenge in scoreChallenges) {
        // get car challenge was created with
        uint64_t context = scoreChallenge.score.context;
        CarModel carModel = [self carModelForContext: context];

        if (carModel == [self carModel]) {
            [challengesWithSameCar addObject: scoreChallenge];
        }
    }
    // continued...
```

Modal Completion

Score Challenge example

```
// load challenges for current player
[GKChallenge loadReceivedChallengesWithCompletionHandler:
^(NSArray *challenges, NSError *error) {

    NSArray *scoreChallenges = [self filterForScoreChallenges: challenges];

    for (GKScoreChallenge *scoreChallenge in scoreChallenges) {
        // get car challenge was created with
        uint64_t context = scoreChallenge.score.context;
        CarModel carModel = [self carModelForContext: context];

        if (carModel == [self carModel]) {
            [challengesWithSameCar addObject: scoreChallenge];
        }
    }
    // continued...
```

Modal Completion

Score Challenge example

```
// load challenges for current player
[GKChallenge loadReceivedChallengesWithCompletionHandler:
^(NSArray *challenges, NSError *error) {

    NSArray *scoreChallenges = [self filterForScoreChallenges: challenges];

    for (GKScoreChallenge *scoreChallenge in scoreChallenges) {
        // get car challenge was created with
        uint64_t context = scoreChallenge.score.context;
        CarModel carModel = [self carModelForContext: context];

        if (carModel == [self carModel]) {
            [challengesWithSameCar addObject: scoreChallenge];
        }
    }
    // continued...
```

Modal Completion

Score Challenge example

```
// load challenges for current player
[GKChallenge loadReceivedChallengesWithCompletionHandler:
^(NSArray *challenges, NSError *error) {

    NSArray *scoreChallenges = [self filterForScoreChallenges: challenges];

    for (GKScoreChallenge *scoreChallenge in scoreChallenges) {
        // get car challenge was created with
        uint64_t context = scoreChallenge.score.context;
        CarModel carModel = [self carModelForContext: context];

        if (carModel == [self carModel]) {
            [challengesWithSameCar addObject: scoreChallenge];
        }
    }
    // continued...
```

Modal Completion

Score Challenge example

```
// load challenges for current player
[GKChallenge loadReceivedChallengesWithCompletionHandler:
^(NSArray *challenges, NSError *error) {

    NSArray *scoreChallenges = [self filterForScoreChallenges: challenges];

    for (GKScoreChallenge *scoreChallenge in scoreChallenges) {
        // get car challenge was created with
        uint64_t context = scoreChallenge.score.context;
        CarModel carModel = [self carModelForContext: context];

        if (carModel == [self carModel]) {
            [challengesWithSameCar addObject: scoreChallenge];
        }
    }
    // continued...
```

Modal Completion

Score Challenge example

```
   [GKScore reportScores: @[currentScore]
  withEligibleChallenges: challengesWithSameCar
   withCompletionHandler: ^(NSError *error) {
       // ...
   }];

}];
```

Modal Completion

Score Challenge example

```
   [GKScore reportScores: @[currentScore]
  withEligibleChallenges: challengesWithSameCar
   withCompletionHandler: ^(NSError *error) {
       // ...
   }];

}];
```

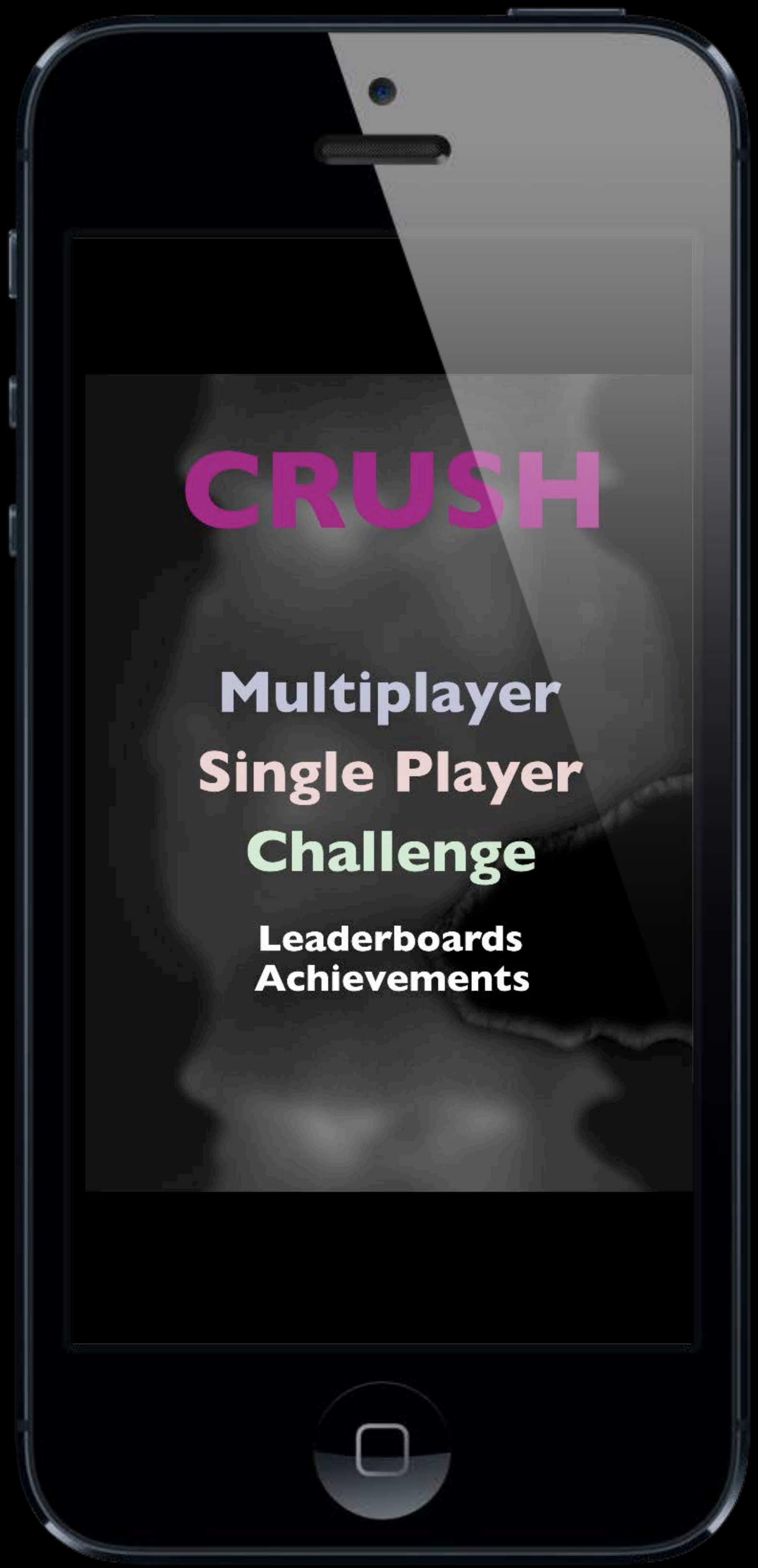
Modal Completion

Score Challenge example

```
    [GKScore reportScores: @[currentScore]
   withEligibleChallenges: challengesWithSameCar
    withCompletionHandler: ^(NSError *error) {
        // ...
    }];

 }];
```

Issuing In-Game

NEW



CRUSH

Multiplayer

Single Player

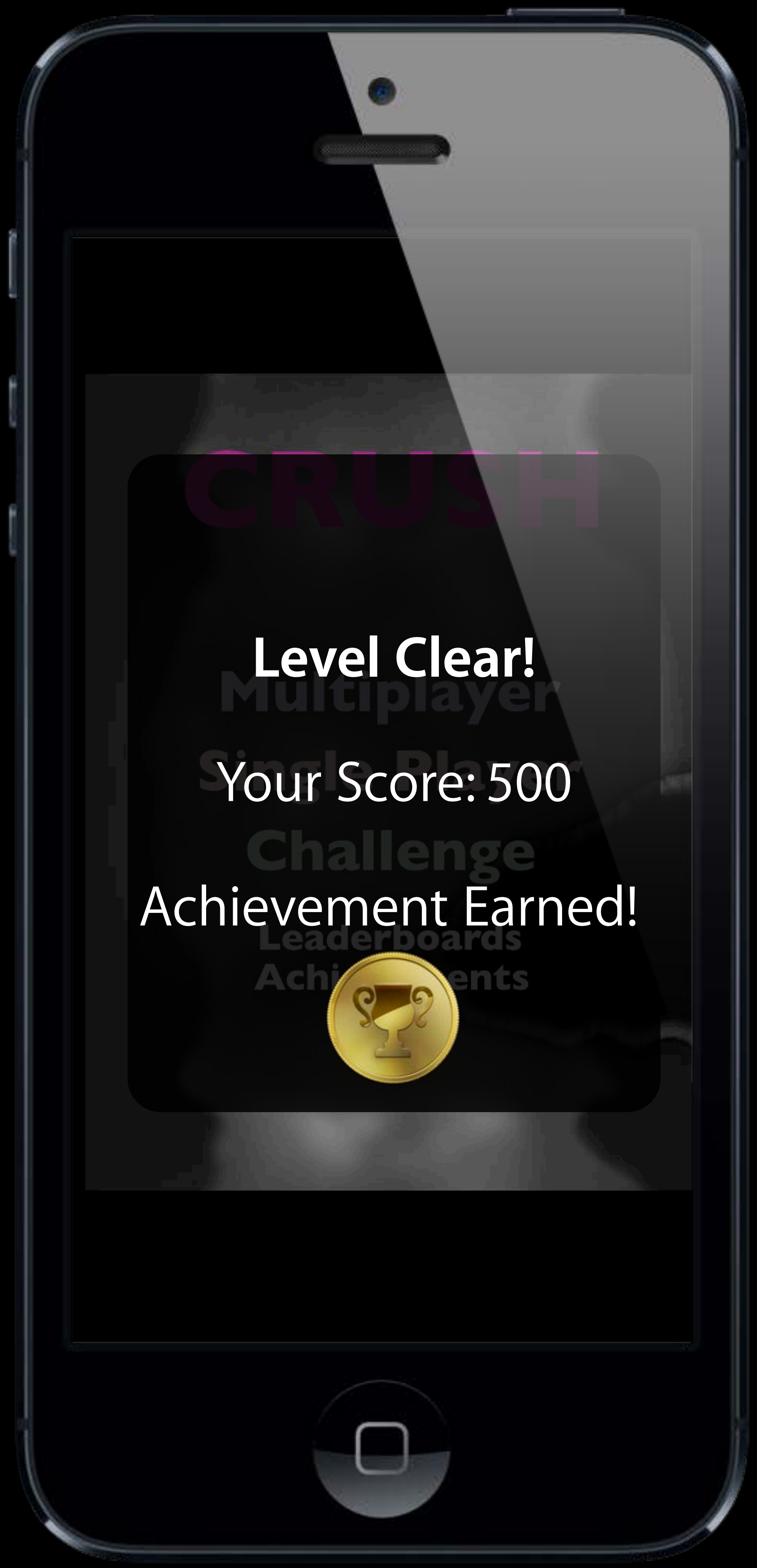
Challenge

Leaderboards

Achievements

Issuing In-Game

NEW



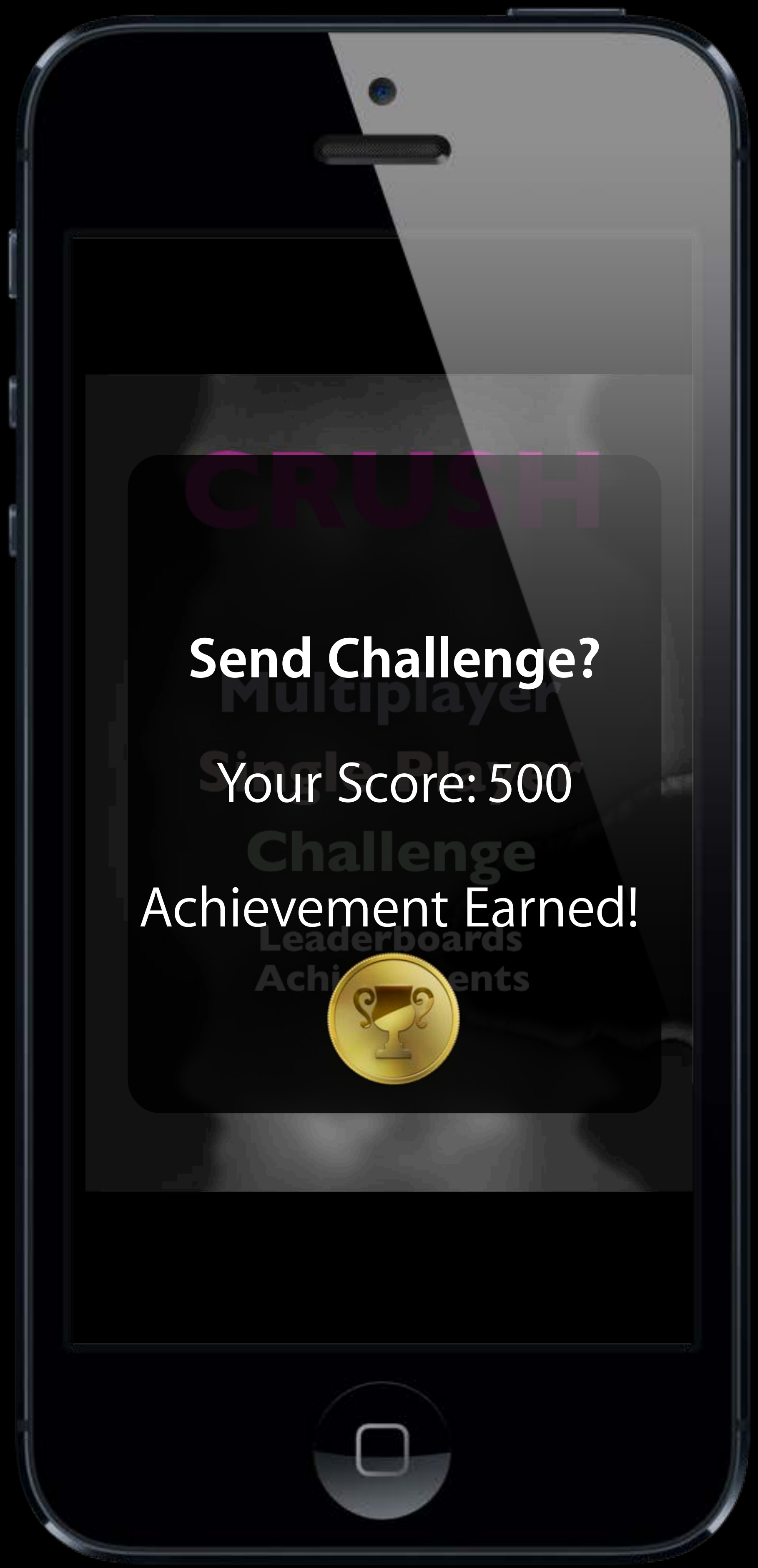
Level Clear!

Your Score: 500

Achievement Earned!

Issuing In-Game

NEW



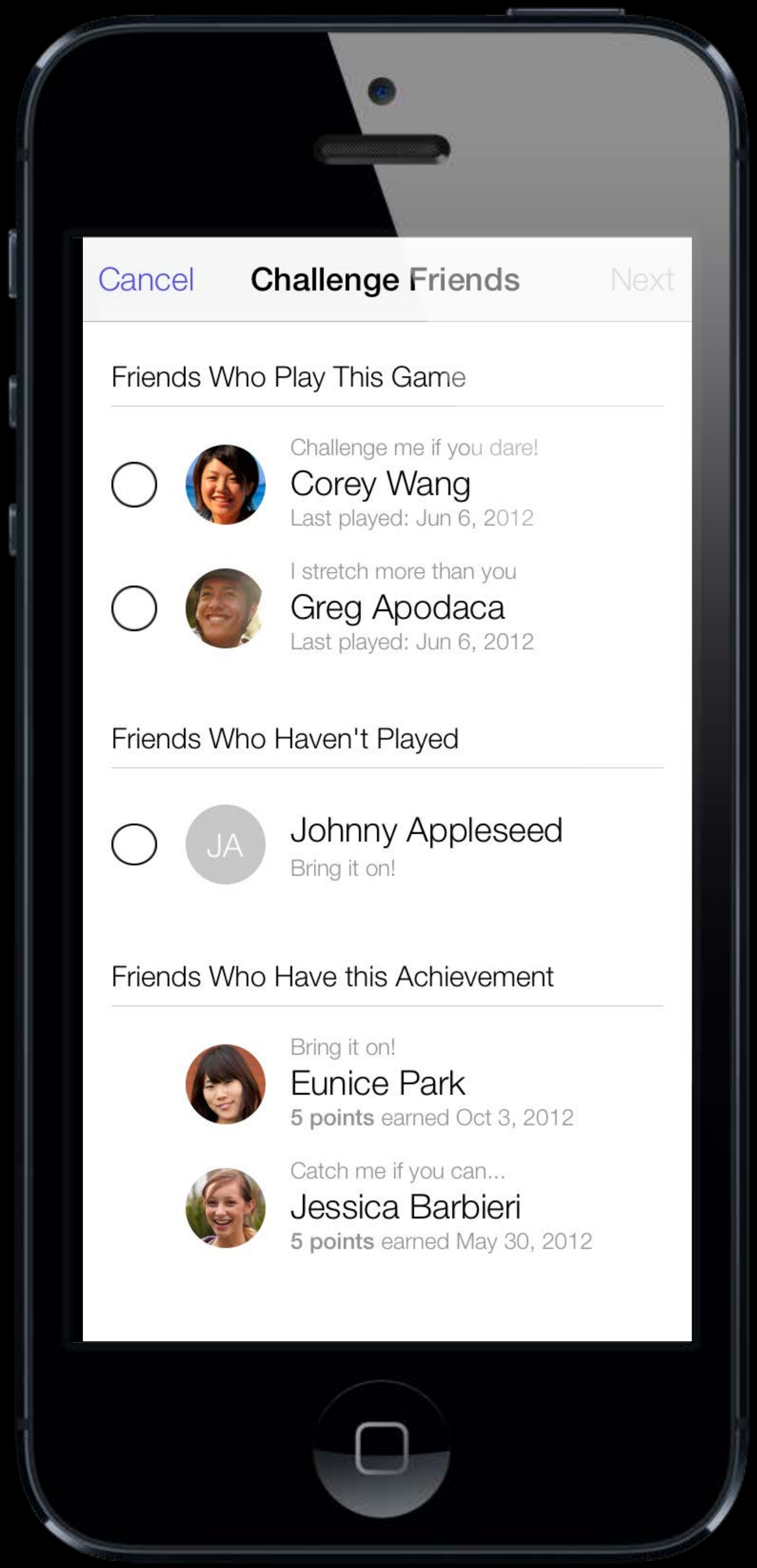
Send Challenge?

Your Score: 500

Achievement Earned!

Issuing In-Game

NEW



Friends Who Play This Game

○ Challenge me if you dare!
Corey Wang
Last played: Jun 6, 2012

○ I stretch more than you
Greg Apodaca
Last played: Jun 6, 2012

Friends Who Haven't Played

○ JA Johnny Appleseed
Bring it on!

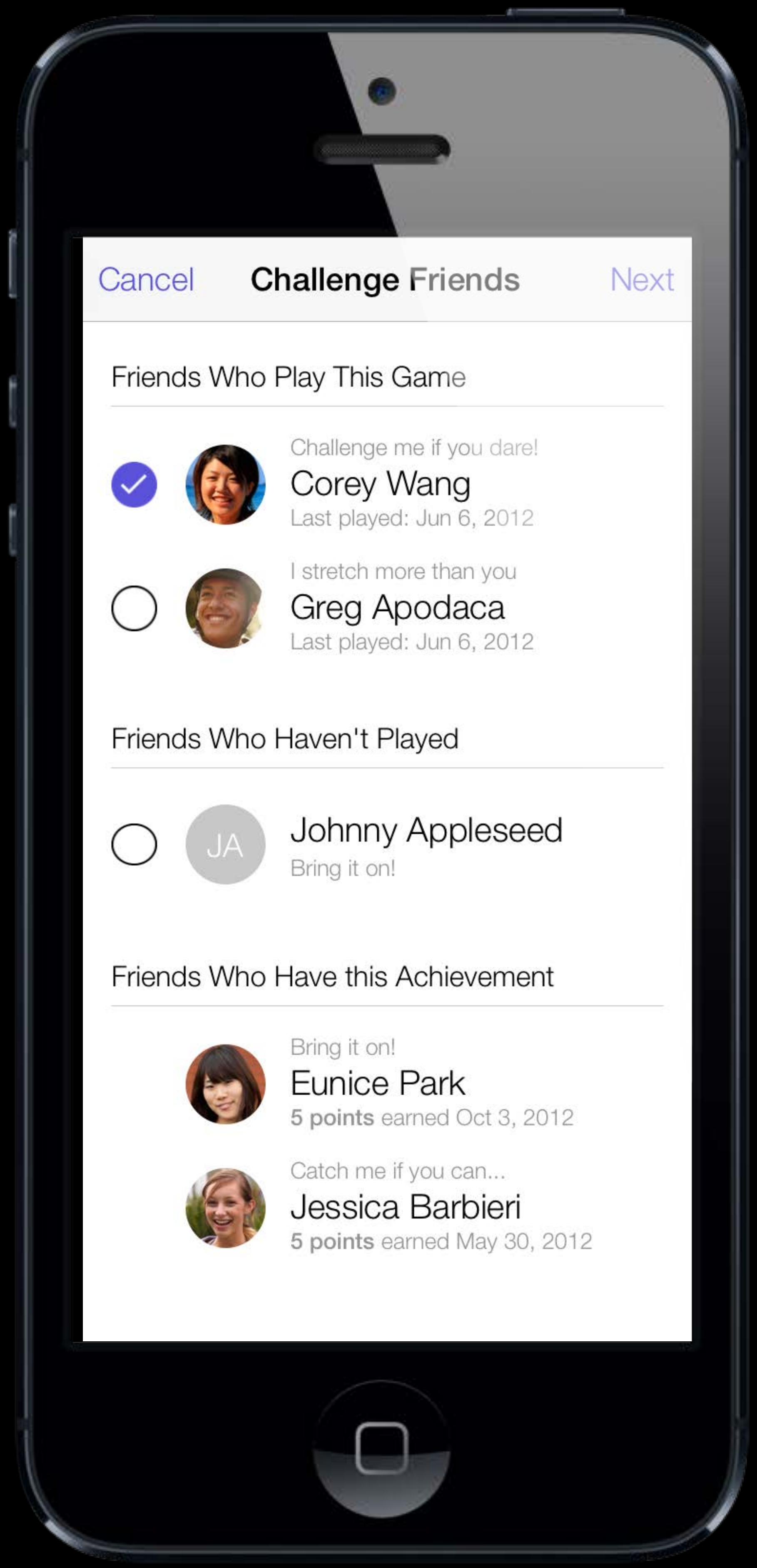
Friends Who Have this Achievement

Bring it on!
Eunice Park
5 points earned Oct 3, 2012

Catch me if you can...
Jessica Barbieri
5 points earned May 30, 2012

Issuing In-Game

NEW



Cancel **Challenge Friends** Next

Friends Who Play This Game

☑ Challenge me if you dare!
Corey Wang
Last played: Jun 6, 2012

○ I stretch more than you
Greg Apodaca
Last played: Jun 6, 2012

Friends Who Haven't Played

○ JA Johnny Appleseed
Bring it on!

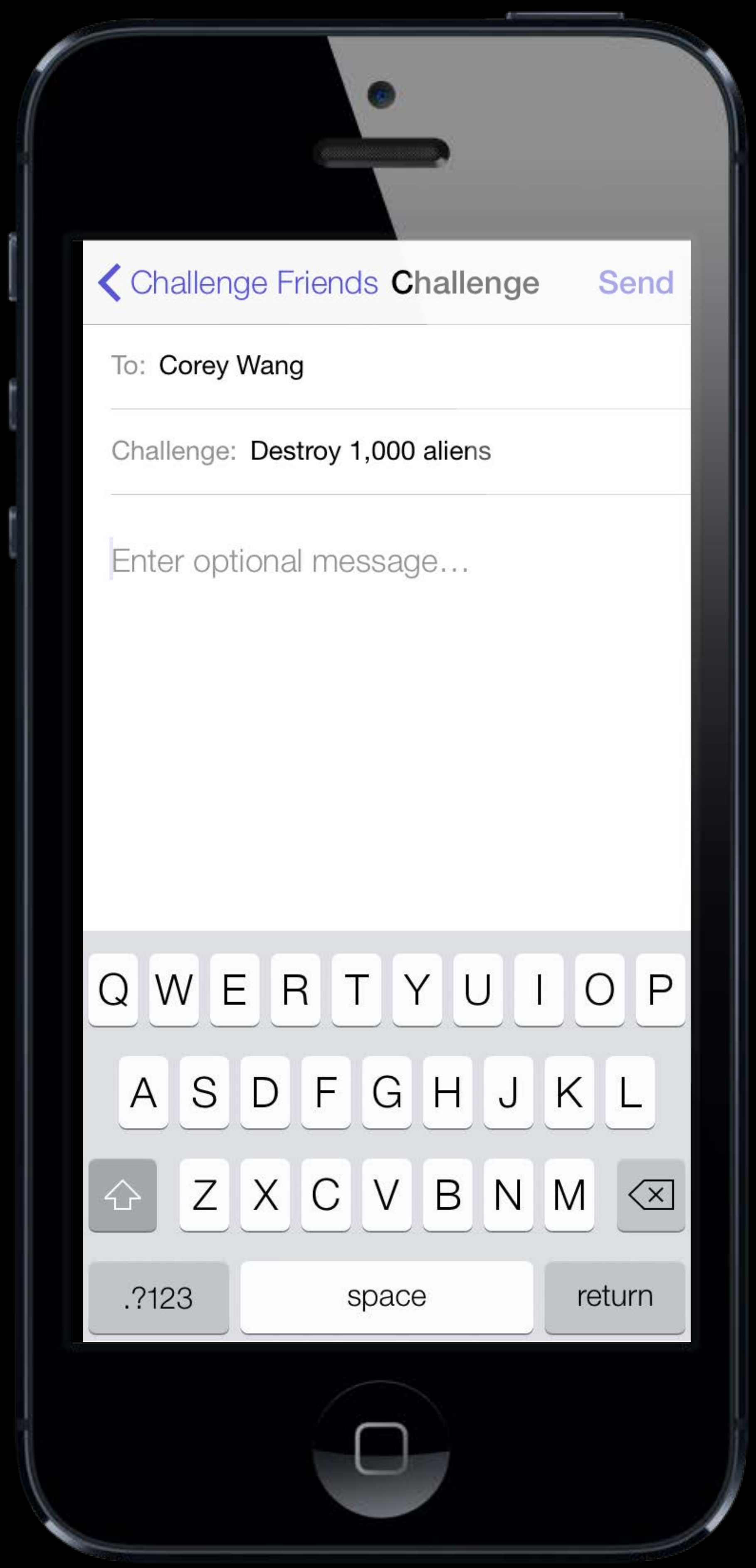
Friends Who Have this Achievement

Bring it on!
Eunice Park
5 points earned Oct 3, 2012

Catch me if you can...
Jessica Barbieri
5 points earned May 30, 2012

Issuing In-Game

NEW



To: Corey Wang

Challenge: Destroy 1,000 aliens

Enter optional message…

Issuing Challenges

Achievement Challenges

NEW

```
// set up the achievement
GKAchievement *myAchievement = [[GKAchievement alloc]
    initWithIdentifier: @”com.mystudio.runandgun”];
myAchievement.percentComplete = 100.0;
```

Issuing Challenges

Achievement Challenges

```
// set up the achievement
GKAchievement *myAchievement = [[GKAchievement alloc]
    initWithIdentifier: @”com.mystudio.runandgun”];
myAchievement.percentComplete = 100.0;
```

Issuing Challenges

Achievement Challenges

```
// get the compose controller
UIViewController *composeVC = [myAchievement
challengeComposeControllerWithPlayers: playersForChallenge
                              message: message
                    completionHandler:
^(UIViewController *controller, BOOL didIssueChallenge, NSArray
*sentPlayerIDs) {

    [controller dismissViewControllerAnimated: YES completion: nil];
    // post processing here
}];

// present
[self.navigationController presentViewController: composeVC animated: YES
completion: nil];
```

Issuing Challenges

Achievement Challenges

```
// get the compose controller
UIViewController *composeVC = [myAchievement
challengeComposeControllerWithPlayers: playersForChallenge
                              message: message
                    completionHandler:
^(UIViewController *controller, BOOL didIssueChallenge, NSArray
*sentPlayerIDs) {

    [controller dismissViewControllerAnimated: YES completion: nil];
    // post processing here
}];

// present
[self.navigationController presentViewController: composeVC animated: YES
completion: nil];
```

Issuing Challenges

Achievement Challenges

```
// get the compose controller
UIViewController *composeVC = [myAchievement
challengeComposeControllerWithPlayers: playersForChallenge
                              message: message
                    completionHandler:
^(UIViewController *controller, BOOL didIssueChallenge, NSArray
*sentPlayerIDs) {

    [controller dismissViewControllerAnimated: YES completion: nil];
    // post processing here
}];

// present
[self.navigationController presentViewController: composeVC animated: YES
completion: nil];
```

Issuing Challenges

Achievement Challenges

```
// get the compose controller
UIViewController *composeVC = [myAchievement
challengeComposeControllerWithPlayers: playersForChallenge
                              message: message
                    completionHandler:
^(UIViewController *controller, BOOL didIssueChallenge, NSArray
*sentPlayerIDs) {

    [controller dismissViewControllerAnimated: YES completion: nil];
    // post processing here
}];

// present
[self.navigationController presentViewController: composeVC animated: YES
completion: nil];
```

Issuing Challenges

Achievement Challenges

```
// get the compose controller
UIViewController *composeVC = [myAchievement
challengeComposeControllerWithPlayers: playersForChallenge
                              message: message
                    completionHandler:
^(UIViewController *controller, BOOL didIssueChallenge, NSArray
*sentPlayerIDs) {

    [controller dismissViewControllerAnimated: YES completion: nil];
    // post processing here
}];

// present
[self.navigationController presentViewController: composeVC animated: YES
completion: nil];
```

Issuing API Changes

```
issueChallengeToPlayers:
                message:
```

```
challengeComposeControllerWithPlayers:
                              message:
                    completionHandler:
```

Issuing API Changes

```
issueChallengeToPlayers:
                message:
```

❌

```
challengeComposeControllerWithPlayers:
                              message:
                    completionHandler:
```

Issuing API Changes

```
issueChallengeToPlayers:
                message:
```

✗

```
challengeComposeControllerWithPlayers:
                              message:
                    completionHandler:
```

✓

Issuing API Changes

```
issueChallengeToPlayers:
                message:
```

✗

```
challengeComposeControllerWithPlayers:
                              message:
                    completionHandler:
```

✓

Challenges Recap

- Modal Challenges for selective completion
- Use challenge compose controller API for issuing
- Deprecated issueChallenge API brings up picker

Event Handling

Event Handling

NEW

- GKLocalPlayer is event producer
- Can have multiple listeners for each event
- Listeners conform to GKLocalPlayerListener
    - GKChallengeListener
    - GKTurnBasedEventListener
    - GKInviteEventListener

Event Handling

Setting the listener

- Adopt any subprotocols of GKLocalPlayerListener
- Register with GKLocalPlayer

```
[[GKLocalPlayer localPlayer] registerListener: self]
```

- Register multiple listeners

Event Handling

Setting the listener

- Adopt any subprotocols of GKLocalPlayerListener
- Register with GKLocalPlayer

```
[[GKLocalPlayer localPlayer] registerListener: self]
```

- Register multiple listeners

Event Handling

Setting the listener

- Adopt any subprotocols of GKLocalPlayerListener
- Register with GKLocalPlayer

```
[[GKLocalPlayer localPlayer] registerListener: self]
```

- Register multiple listeners

GKChallengeListener

```
- (void)player: (GKPlayer *)player
wantsToPlayChallenge: (GKChallenge *)challenge;

- (void)player: (GKPlayer *)player
didReceiveChallenge: (GKChallenge *)challenge;

- (void)player: (GKPlayer *)player
didCompleteChallenge: (GKChallenge *)challenge
issuedByFriend: (GKPlayer *)friendPlayer;

- (void)player: (GKPlayer *)player
issuedChallengeWasCompleted: (GKChallenge *)challenge
byFriend: (GKPlayer *)friendPlayer;
```

Selecting Challenges

```
- (void)player: (GKPlayer *)player wantsToPlayChallenge: (GKChallenge
*)challenge
{
    // switch to challenge mode for challenge
    [self startChallengeModeForChallenge: challenge];
}
```

Receiving Challenges

```
- (void)player: (GKPlayer *)player didReceiveChallenge: (GKChallenge
*)challenge
{
    // display custom UI for challenge
    [self presentChallengeInGame: challenge];
}
```

Completed Challenges

```
- (void)player: (GKPlayer *)player didCompleteChallenge: (GKChallenge
*)challenge issuedByFriend: (GKPlayer *)friendPlayer
{
    // reward player for completing a challenge
    [self displayCongratsBanner];
}
```

Completed Challenges

```
– (void)player: (GKPlayer *)player issuedChallengeWasCompleted:
(GKChallenge *)challenge byFriend: (GKPlayer *)friendPlayer
{
    // player’s friend beat a challenge they sent
    [self playSadTrombone];
}
```

GKLocalPlayerListener

- Deprecated
  - GKChallengeEventHandler
  - GKTurnBasedEventHandler
  - GKMatchmaker inviteHandler property
- Affects multiplayer too
  - Check out multiplayer session tomorrow

What We’ve Learned

- Leaderboard Sets
- Most Recent Score Leaderboards
- Security
- Challenges
- New GKLocalPlayerListener API

More Information

Allan Schaffer
Graphics and Game Technologies Evangelist
aschaffer@apple.com

Documentation
Game Center for Developers
http://developer.apple.com/game-center

Apple Developer Forums
http://devforums.apple.com

Related Sessions

What’s New in iTunes Connect	Pacific Heights Thursday 10:15AM	🟩
Turn-Based Gaming with Game Center	Mission Thursday 11:30AM	🟨

Labs

Game Center Lab	Graphics and Games Lab B Thursday 12:45PM

WWDC2013