



Game Center Techniques, Part 1

Get Your Game On

Gabriel Belinsky
Senior Software Engineer

Game Center

- Friends
- Leaderboards
- Achievements
- Multiplayer gaming



What You'll Learn

- Game Center API basics
- Authenticate with Game Center
- Remember players across different play sessions
- Report a high score
- Access and show a Leaderboard in game
- Report an achievement
- Access and show achievements in game
- Reset user's achievements

Game Center Overview



General Services

Authentication

Friends

Leaderboards

Achievements

Blocks-Based API

Definition of blocks

```
main()
```

```
{
```

```
int x = 123;  
int y = 456;
```

← Captures Enclosing Variables

```
void (^myBlock)(int) = ^(int z) {  
    printf("%d %d %d\n", x, y, z);  
}
```

← Anonymous function

Blocks can be declared like any C variable

```
myBlock(789);
```

← Outputs 123 456 789

```
}
```

Blocks-Based API

API usage

```
[gameKitObject doActionWithCompletionHandler:^(id result, NSError *error)
{
    if (result) {
        // process the result
    }
    if (error) {
        // process the error
    }
}];
```

GameKit Errors

GKError.h

- GKErrorCancelled
- GKErrorCommunicationsFailure
- GKErrorInvalidCredentials
- GKErrorNotAuthenticated
- GKErrorAuthenticationInProgress
- GKErrorScoreNotSet
- GKErrorParentalControlsBlocked
- GKErrorMatchRequestInvalid

General Services

Authentication

Friends

Leaderboards

Achievements

GKLocalPlayer

Represents current player on the device

- Holds current player info
 - playerId
 - alias
 - underage
- Performs authentication
- Access friends list

GKLocalPlayer

Authentication

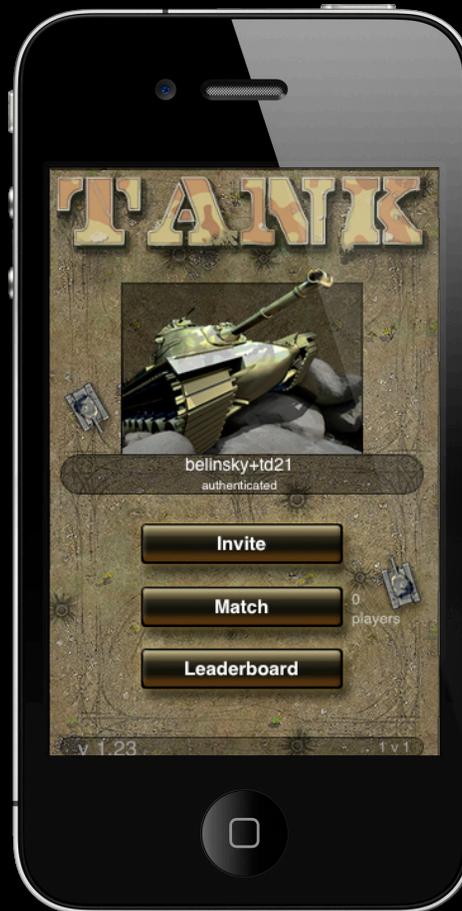
- Required to use the rest of Game Center

```
GKLocalPlayer*  localPlayer = [GKLocalPlayer localPlayer];

[localPlayer authenticateWithCompletionHandler:^(NSError *error) {
    if (!error) {
        self.gameCenterFeaturesEnabled = YES;
    } else {
        self.gameCenterFeaturesEnabled = NO;
    }
}];
```

Authentication

Screenshots



General Services

Authentication

Friends

Leaderboards

Achievements

GKPlayer

Properties

- playerId
 - Persistent identifier
 - Keep track of players over time
 - Associate data with players
- alias
 - Display name for player

Friends

Getting friends' list

```
GKLocalPlayer*    localPlayer = [GKLocalPlayer localPlayer];

[localPlayer loadFriendsWithCompletionHandler:^(NSArray *friends, NSError *error) {
    if (friends) {
        // friends array contains a GKPlayer for each friend of localPlayer
    }
    if (error) {
        // handle error
    }
}];
```

What to Do with the Friends' List

- Attach custom data to players on your own server
- Extend Game Center functionalities
 - Virtual goods
 - Cross-session messaging
- Use the social graph as part of your gameplay
 - Let your users visit each other's virtual places
 - Use the connections to enhance their virtual character

General Services

Authentication

Friends

Leaderboards

Achievements

Leaderboard

Concepts

- A list of high scores earned in a game
- Filters for friends and time
- Can be broken out by category
- Optional global Leaderboard



Leaderboard

Steps

- Server-side setup
- Report a score
- Show Leaderboard to user
- Query Leaderboard data

Leaderboard

Server-side setup

- Done in iTunes Connect
- Enable Leaderboard for your game
- Set up your Leaderboard
 - Leaderboard identifier
 - Sort order
 - Add languages
 - Score format
 - Optional score format suffix
- More languages = more sales

[Store](#)[Mac](#)[iPod + iTunes](#)[iPhone](#)[Downloads](#)[Support](#)

iTunes Connect

Edit Application

Leaderboard Details

A leaderboard is the place for your users to view the top scores by all Game Center players of your app. Once a leaderboard has gone live for this version of your app, it cannot be removed.

Leaderboard ID : com.apple.touchfighter.leaderboard 

Sorting : Ascending Descending 

Leaderboard Format [Add New Language](#)

Language	Score Format	Score Format Suffix	
English	Integer (214,000,529)	Gold Coins	Delete
Spanish	Integer (214,000,529)	Monedas de Oro	Delete

Leaderboard Formatting

Example

Submitted Value	Selected Formatter	Formatted Value
123456	Integer (214,000,529)	123,456
792	Elapsed Time (hours, minutes)	13:12
3723	Elapsed Time (hours, minutes, seconds)	1:02:03
1042	Money (\$10,000,012)	\$1,042

Reporting a Score

- Allocate a GKScore
 - Set value
 - Set category if using multiple Leaderboards
- Report score
 - Developer responsible for retry on network errors

GKScore

Properties

- value
- formattedValue
- category
- date
- player
- rank

Reporting a Score

```
GKScore* myPlayerScore = [[[GKScore alloc] init] autorelease];
myPlayerScore.value = 12345;

[myPlayerScore reportScoreWithCompletionHandler:^(NSError *error)
{
    if (error) {
        NSData* archivedScore = [NSKeyedArchiver
                                archivedDataWithRootObject:myPlayerScore];
        [self saveScoreToSendLater:archivedScore];
    } else {
        // the score was submitted successfully
    }
}];
```

Leaderboard Standard UI

```
- (void)showLeaderboard
{
    GKLeaderboardViewController*
    myLeaderboardViewController = [[[GKLeaderboardViewController alloc]
        initWithTimeScope:GKLeaderboardTimeScopeWeek
        playerScope:GKLeaderboardPlayerScopeFriendsOnly
        ] autorelease];

    [myLeaderboardViewController setLeaderboardDelegate: self];
    [self presentModalViewController: myLeaderboardViewController];
}

- (void)leaderboardDidPressDismiss
{
    [self dismissModalViewControllerAnimated:NO];
}
```

Querying a Leaderboard

- Allocate a GKLeaderboard
 - Set time scope (today/week/all time)
 - Set player scope (friends/everyone)
 - Set range of ranks desired
 - Set category if using multiple leaderboards
- Returns an NSArray of GKScores which contain the following properties:
 - player (a GKPlayer object)
 - rank
 - value
 - formattedValue
 - date

Querying a Leaderboard

```
GKLeaderboard* myLeaderboard = [[[GKLeaderboard alloc] init] autorelease];

// interested in friends scores over the past week
myLeaderboard.timeScope = GKLeaderboardTimeScopeWeek;
myLeaderboard.playerScope = GKLeaderboardPlayerScopeFriendsOnly;
myLeaderboard.range = NSMakeRange(1,25);

[myLeaderboard
 loadScoresWithCompletionHandler:^(NSArray *scores, NSError *error) {
    if (scores) {
        //scores now contains all the scores received per the query
        for (GKScore *score in scores) {
            // looping through all the scores in the array
            // display in custom UI
        }
    }
    if (error) {
        // handle error
    }
}];
```

Leaderboard Screenshots



General Services

Authentication

Friends

Leaderboards

Achievements

Achievements

Concepts

- Something of significance that is accomplished by a player
 - Completed a level within 60 seconds
 - Looted all the treasure in the dungeon
 - Checking out our labs tomorrow
- Increases player engagement in your game
- Encourages online and offline competition

Coming
Soon



Achievements

Steps

- Server-side setup
- Reporting achievements
- Getting achievement descriptions
- Querying achievements
- Show achievements to user
 - Standard UI
 - Custom UI
- Resetting achievements list



Achievements

Server-side setup

A circular logo with a dark background and a light border. The text "Coming Soon" is written in a white, sans-serif font, with "Coming" on the top line and "Soon" on the bottom line.

- Done with iTunes Connect
 - Enable achievements for your game in iTunes Connect
 - For each achievement:
 - Enter the achievement ID (must be unique, inverse DNS is perfect)
 - Enter the achievement title
 - Enter the description before the user achieves it
 - Enter the description after the user achieves it
 - Enter a point value
 - Select the hidden flag if you do not want the user see that achievement before you unhide it
 - Upload the icon you want to attach to this achievement

Achievements

GKAchievementDescription

- This is the list of achievements for your game
 - Unique ID
 - Title
 - Icon
 - Description if not achieved
 - Description if achieved
 - Points
 - Flag to hide the achievement



Achievements

GKAchievement

- This is the list of achievements attached to GKLocalPlayer
 - Unique ID
 - Percentage completed
 - Hidden flag
 - Completed flag
 - Date of report
 - Points

Reporting an Achievement

- Allocate a GKAchievement for the achievement ID
 - Set percent complete
- Developer is responsible for retry on network errors

Reporting an Achievement

```
GKAchievement* myAchievement = [[[GKAchievement alloc]
    initWithIdentifier:@"com.myCompany.myGame.achievementX"] autorelease];

myAchievement.percentComplete = 50.0;

[myAchievement reportAchievementWithCompletionHandler:^(NSError *error) {
    if (error) {
        NSData* archivedAchievement = [NSKeyedArchiver
            archivedDataWithRootObject:myAchievement];
        [self saveAchievementToSendLater:archivedAchievement];
    } else {
        // the achievement was submitted successfully
        // Note: reporting an achievement will unhide it
    }
}];
```

Achievements Standard UI

Implement GKAchievementViewControllerDelegate

```
- (void)showAchievements
{
    GKAchievementViewController* myAchievementsViewController =
        [[[GKAchievementViewController alloc] init] autorelease];
    [myAchievementsViewController setAchievementViewControllerDelegate: self];

    [self presentModalViewController:myAchievementsViewController];
}
```

```
- (void)achievementsDidPressDismiss
{
    [self dismissModalViewControllerAnimated:YES];
}
```

Getting Achievements Descriptions

```
@property (retain) NSMutableDictionary *achievementDescriptionTable;

[GKAchievementDescription loadAchievementDescriptionsWithCompletionHandler:
 ^(NSArray *descriptions, NSError *error) {
    if (descriptions) {
        for (GKAchievementDescription *description in descriptions) {
            [self.achievementDescriptionTable
             setObject:description
             forKey:description.identifier];
        }
    }
    if (error) {
        // handle error
    }
}];
```

Querying Achievements

```
@property (retain) NSMutableDictionary *achievementDescriptionTable;

- (void)logAchievements
{
    [GKAchievement loadAchievementsWithCompletionHandler:
     ^(NSArray *achievements, NSError *error) {
        if (achievements) {
            for (GKAchievement *achievement in achievements) {
                GKAchievementDescription *description;
                description = [self.achievementDescriptionTable
                               objectForKey:achievement.identifier];

                NSLog(@"achievement: %d/%d - %@", achievement.points,
                                                         description.maximumPoints,
                                                         description.title);
            }
        }
        if (error) {
            // handle error
        }
    }];
}
```

Resetting User's Achievements

```
[GKAchievement resetAchievementsWithCompletionHandler:^(NSError *error) {  
    if (error) {  
        // Achievements list was not reset. Process error  
    } else {  
        // the user's achievements list was reset  
    }  
}];
```

Achievements Screenshot



Summary

- Game Center is a blocks-based API
- Asynchronous blocks API
- You must authenticate before doing anything else
- Report a high score
- Access and show a Leaderboard in game
- Report an achievement
- Access and show achievements in game
- Reset a user's achievements
- Achievements will be available later this year

More Information

Allan Schaffer

Graphics and Game Technology Evangelist

ashaffer@apple.com

Documentation

iPhone Dev Center

<http://developer.apple.com/iphone>

Related Sessions

Introduction to Game Center

Pacific Heights
Tuesday 2:00PM

Game Center Techniques, Part 2

Marina
Tuesday 4:30PM

What's New in Foundation for iOS 4

Pacific Heights
Tuesday 10:15AM

Introducing Blocks and Grand Central Dispatch on iPhone

Russian Hill
Wednesday 11:30PM

For more on blocks go to Apple Developer Connection at:

<http://developer.apple.com>

Labs

Game Center Lab

Lab B
Wednesday 2:00PM

Game Center Lab

Lab D
Friday 12:45PM

Q&A



