

ユーザー管理サンプル

*\* このサンプルは、2016年3月のXbox One XDK以降と互換性があります。*

# 概要

このサンプルは、元となるなるシステム内で、ユーザーとコントローラーが追加、削除、およびそれらのペアリングが変更されたときに受信されるUserイベントとControllerイベントに、従います。また、アクティブユーザーを取得し、停止から再開するときにそのユーザーを更新する方法も示します。

# サンプルの使用



|  |  |
| --- | --- |
| 作用 | ゲームパッド |
| アカウント ピッカーを表示 | メニューボタン |
| コントローラを一覧表示します | X ボタン |
| このコントローラを一覧表示します | 左トリガー+ Xボタン |
| ユーザーを一覧表示します | Y ボタン |
| アクティブユーザーの設定/リセット | A/Bボタン |
| サインアウト延期時間の調整（秒） | DPadの左/右 |
| 終了 | 表示ボタン |

# 実装上の注意

ラムダ関数内に実装されているすべてのUserおよびControllerイベント処理コードについては、UserManagement.cppのInitialize（）メソッドを参照してください。スコープロックは、各イベントハンドラからのすべてのコンソール出力をまとめて保持します。

また、UserManagement.cppのOnResuming（）メソッドにProcess Lifetime Management（PLM）コードを表示して、停止状態から再開するときに失効したアクティブユーザーを確認して処理する方法を学ぶこともできます。

「アクティブユーザー」について

このサンプルは、ユーザーにゲームパッドのAボタンを押すように指示することで、「**アクティブユーザー**」を取得するための「エンゲージメントプロンプト」をシミュレートします。このサンプルでは、現在アクティブなユーザーが画面の左上に表示されています。

これは、あるタイトルがユーザープロファイル関連のアクション（例えば、ユーザーのリッチなプレゼンスを描く、ユーザーの進捗や設定を保存する、業績を解除するなど）を実行する場合、そのユーザーのプロファイルに対して何らかのアクションが実行される前に、そのタイトルがアクティブユーザーを指し示している必要があるというXboxの要件を示すのに役立ちます。アクティブユーザーが誰であるかを判断する方法は複数あります。ほとんどのゲームでは、起動時にユーザー獲得画面を使って続行するためのボタンを押すように促され、このプロセスを通じてゲームはアクティブなユーザーをとらえることができます。サインインしているユーザーが1人だけの場合は、そのユーザーがアクティブユーザーであると見なすことができます。接続されているコントローラが1つだけの場合は、それが使用されていて、それとペアになっているユーザに対してクエリが実行されていると見なすことができます。必要に応じて、ゲームは**ShowAccountPickerAsync**を使用して、ユーザーを選ぶように促すことができます。最初のユーザーの獲得後、複数のユーザーがゲームに参加できる場合は、ShowAccountPickerAsyncへの追加呼び出しを使用して追加されるユーザーを追加できます。

また、比較と実験のために、このサンプルではXbox **CurrentUser**プロパティも画面上部に表示されています。Current UserはXboxアプリケーションモデルの一部であり、ゲームが起動または再開（サスペンドモードから）されたとき、どのプロファイルがHomeでアクティブなプロファイルだったかを示します。

このプロパティと**CurrentUserChanged**イベントを介してアクティブユーザーを追跡することは、ユーザー管理を実装するもう1つの方法ですが、通常はゲームにはお勧めできません。このプロパティは、ゲームが予測するタイミングに更新されるとは限りません。たとえば、現在のXbox OSでは、ユーザーがゲームパッドのXboxボタンを押してガイドを表示して[プロファイルの切り替え]を選択すると、コントローラーペアリングを変更できますが、現在のユーザーは変更されません。この理由とその他の理由から、ゲームにとって最適であり、Xboxのユーザー管理要件に準拠している独自の「アクティブユーザー」の概念を実装することをお勧めします。

これらの概念については、XDKのドキュメントおよびXboxの要件に関するドキュメントで詳しく説明しています（特に**XR-112**を参照: 初期ユーザーとコントローラーの確定、および**XR-113**：Active User Management、および下記の参考文献に記載されている他のドキュメント）

サンプルの制限

このサンプルでは、以下の制限事項に注意してください。

1. アクティブユーザーとペアになっている「アクティブコントローラ」は追跡されません
2. Kinectはアクティブユーザーの設定には使用されません

# 試すべきこと

以下の各アクションは、ユーザー管理に関連するいくつかの重要なシナリオを説明するのに役立ちます。  
画面上のログに表示される変更とイベントを確認してください。

1. **アクション：**ゲームパッドのXboxボタンを使ってガイドを開き、プロファイルの切り替えを選択します。  
   **観察：**さまざまなユーザーおよびコントローラのイベントが発生する可能性がありますが、現在のユーザーは同じままなので注意が必要です。
2. **アクション：**Xboxボタンを使用してGo Home（Dev HomeまたはRetail Home）に移動し、ガイドを表示してプロファイルを切り替えてから、サンプルを再開します。  
   **観察：**現在のユーザーはこのシナリオで更新されます。アクティブユーザーを消去してエンゲージメントプロンプトを表示するかどうかは、タイトル次第です。このサンプルは、アクティブユーザーのアクティブコントローラを追跡しません。
3. **アクション：**サインインしてアクティブユーザーを確定したら、他の場所でサインインしてシングルポイントオブプレゼンス（SPOP）を確認します。  
   **観察：**次の一連のイベントが発生します。
   1. SignOutStartedイベント（例えばサインアウト延期オブジェクトを保持している間にゲーム状態の保存を開始できるところです）
   2. サインアウトの延期が行われます（このサンプルのデフォルトの期間は5秒に設定されています）。
   3. ControllerPairingChangedイベントは、ユーザーペアリングをコントローラーから削除します
   4. UserRemovedイベント
   5. SignOutCompletedイベント（これはアクティブユーザがリセットされるところです）
4. **アクション：**サインインしてアクティブユーザーを確立したら、Xbox OneマネージャーまたはDev Homeを使用してサンプルを停止し、アクティブユーザーをサインアウトしてから再開します。  
   **観察：**再開すると、サンプルは失効したアクティブユーザーを見つけ、そのユーザーがサインインしていない状態でアクティブユーザーをクリアしし、エンゲージメントプロンプトを再度表示します。その間、Current User プロパティが変更されていないこと、およびSignOutイベントが受信されていないことを観察します。
5. **アクション：**サインインしてアクティブユーザーを確立して、Xbox OneマネージャーまたはDev Homeを使用してサンプルを中断し、アクティブユーザーをサインアウトし、サンプルがまだ中断されている間にそのユーザーをサインバック(サインインの復帰)して再開します。  
   **観察：**再開すると、サンプルは失効したアクティブユーザーを見つけますが、そのユーザーはサインインしているため、アクティブユーザーオブジェクトを更新できます。SignOut イベントが受信されなかったことにも注意してください。

# よくあるご質問

1. コンソールに1つしか接続していないのに、2つ（またはそれ以上）のゲームパッドコントローラが表示されるのはなぜですか？  
   **答え**: Xbox One マネージャーを実行していてdevkitに接続している場合は、ゲームパッドとしてコントローラ一覧に表示されます。Xbox One マネージャーを終了してみると、コントローラ削除イベントが表示されるはずです。
2. ユーザーやコントローラーのイベントのイベントが発生しているのを見つけられませんでした。何が起こったのですか？  
   **答え:** これはまれに起こることがあります。コンソールを再起動することにより、いずれかのシステムサービスの問題を解決しようとしてください。また、Xbox One マネージャーの[問題を報告]ツールを使用してフィードバックを送信して報告することをお勧めします。

# 資料

* XR-112: 初期ユーザとコントローラの確立
* XR-113: アクティブユーザー管理
* XR-115: ユーザーとコントローラの追加と削除
* XR-116: 一時停止モードおよび制約モードから再開するときのユーザーとコントローラの処理
* [Xfest 2014](https://developer.xboxlive.com/ja-jp/platform/documentlibrary/events/Pages/Xfest2014.aspx) トーク：「プレーヤーと入力の管理: 誰が私のゲームをプレイしていますか？」  
  [Video](http://go.microsoft.com/?linkid=9845004) | [.pptx](https://developer.xboxlive.com/_layouts/xna/download.ashx?file=InputAudioSG_Managing_Players_and_Input_Who_is_Playing_My_Game.pptx&folder=platform\xfest2014)
* [ホワイトペーパー](https://developer.xboxlive.com/ja-jp/platform/development/education/Pages/WhitePapers.aspx)：「[ユーザー、コントローラー、ペアリング - Xbox OneのID](http://aka.ms/9838344)」

# 既知の問題点

なし

# 更新履歴

**初回リリース：**Xbox Oneの発売

**2018年3月更新：**

* 新しいサンプルフレームワークを使用
* UIの見直し
* アクティブユーザーの概念を実装。エンゲージメントプロンプトを通じて取得され、ユーザーがサインアウトしたときやサンプルが中断から再開したときに自動的に更新されます。
* 失効したアクティブユーザをチェックし、必要に応じて更新します
* システムによって設定されたCurrentUserプロパティを追跡します。
* *複数* コントローラーからの個々のサインインとペアリングをサポート
* すべてのコントローラーに関する情報の表示に加えて、ボタンが押されたコントローラーに関する情報（例：現在のコントローラー）の表示をサポートします。
* ユーザーイベントやコントローライベントが発生したときに配置できるように、視界やPLMイベントなど、より多くのイベントがオンスクリーンコンソールに記録されます。

# プライバシーステートメント

サンプルをコンパイルして実行すると、サンプルの実行可能ファイルの名前がMicrosoftに送信され、サンプルの使用状況の追跡に役立ちます。このデータ収集を無効にするには、Main.cppの「Sample Usage Telemetry」というラベルの付いたコードブロックを削除します。

マイクロソフトのプライバシーポリシー全般に関する詳細については、 [Microsoftのプライバシーステートメント](https://privacy.microsoft.com/ja-jp/privacystatement/)をご参照ください。