

IBM Worklight Foundation V6.2.0 **入門**

オフライン作業



商標

- IBM、IBM ロゴ、ibm.com および Worklight は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。
- Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。
- この資料は、事前に IBM の書面による許可を得ずにその一部または全部を複製することは禁じられています。

IBM® について

- <http://www.ibm.com/ibm/us/en/> を参照してください。

アジェンダ

- オフライン・モードでの作業
- アクティブ検出 - メソッドの使用
- パッシブ検出 - オフライン・イベントおよびオンライン・イベント
- 追加のメソッド
- フォアグラウンド・イベント
- Worklight ハートビート
- サンプル

オフライン・モードでの作業

- IBM Worklight® Foundation を使用することにより、アプリケーション接続障害を検出して、行動方針を決定することができます。
- 以下の2つの方法で、アプリケーションのオフライン/オンライン状態への移行を検出することができます。
 - サーバー・ベースのプロシージャラーの呼び出し時に明示的に
 - JavaScript™ イベント・リスナーを使用して暗黙的に
- オフライン状況およびオンライン状況に対してカスタムのアプリケーション動作を定義できます。
- アプリケーション内でのオフライン状態またはオンライン状態を維持し、アプリケーションが失敗したサーバーへの接続試行からリカバリできるようにする必要があります。
 - 例えば、新しいユーザーでアプリケーションにログインする前に、あるいは、新しいユーザーでサーバーにアクセスする前に、アプリケーションは、サーバーがログアウトの成功を受け取っていることを確認する必要があります。

アジェンダ

- オフライン・モードでの作業
- アクティブ検出 - メソッドの使用
- パッシブ検出 - オフライン・イベントおよびオンライン・イベント
- 追加のメソッド
- フォアグラウンド・イベント
- Worklight ハートビート
- サンプル

アクティブ検出 - メソッドの使用

- アプリケーション・コード内の以下の2つの場所で、接続が失われたことを検出できます。
 - アプリケーションの初期化 – `WL.Client.init()` メソッド (通常は `initOptions.js` ファイルから呼び出される)
 - アダプター・プロシージャラーの呼び出し – `WL.Client.invokeProcedure()` メソッド
- 接続障害検出をどちらの場所に追加する場合でも、`onConnectionFailure` プロパティを追加し、接続が失敗した場合に呼び出すコールバック関数を指定します。

```
var wlInitOptions = {  
  onConnectionFailure: function (data) {  
    connectionFailure(data);  
  },  
  WL.Client.invokeProcedure(invocationData, {  
    onSuccess: successHandlerFunction,  
    onConnectionFailure: connectionFailure,  
    timeout: 1000  
  });  
};
```

アジェンダ

- オフライン・モードでの作業
- アクティブ検出 - メソッドの使用
- **パッシブ検出 - オフライン・イベントおよびオンライン・イベント**
- 追加のメソッド
- フォアグラウンド・イベント
- Worklight ハートビート
- サンプル

パッシブ検出 - オフライン・イベントおよび オンライン・イベント

- Worklight フレームワークは Worklight Server にアクセスしようとするたびに、アプリケーションがオフライン状況からオンライン状況 (またはその逆) に切り替わったことを検出する可能性があります。
- どちらの場合も、JavaScript イベントが発生します。
 - `WL.Events.WORKLIGHT_IS_DISCONNECTED` イベントは、Worklight Server への接続が失敗したときに発生します。
 - `WL.Events.WORKLIGHT_IS_CONNECTED` イベントは、Worklight Server への接続が復元されたときに発生します。
- これらのイベントにイベント・リスナーを追加し、イベントを処理するためのコールバック関数を指定することができます。

```
document.addEventListener(WL.Events.WORKLIGHT_IS_CONNECTED, connectDetected,  
false);
```

```
document.addEventListener(WL.Events.WORKLIGHT_IS_DISCONNECTED,  
disconnectDetected, false);
```

- **注** : `WL.Events.WORKLIGHT_IS_DISCONNECTED` と `WL.Events.WORKLIGHT_IS_CONNECTED` は、ストリングではなく名前空間定数です。

アジェンダ

- オフライン・モードでの作業
- アクティブ検出 - メソッドの使用
- パッシブ検出 - オフライン・イベントおよびオンライン・イベント
- 追加のメソッド
- フォアグラウンド・イベント
- Worklight ハートビート
- サンプル

追加のメソッド (1/2)

- オンライン開発およびオフライン開発を単純化するために、Worklight フレームワークによって追加のメソッドが提供されます。
 - `WL.Client.connect (options)` – Worklight Server への接続の確立を試行して、オンライン・モードに戻ります。options は、以下のキーを含むオブジェクトです。
 - `onSuccess` – サーバー接続が確立されたときに呼び出すコールバック関数
 - `onFailure` – サーバー接続が失敗したときに呼び出すコールバック関数
 - `timeout` – サーバー応答を待機するミリ秒数で、この時間を経過するとタイムアウトで要求は失敗します

追加のメソッド (2/2)

- `WL.Device.getNetworkInfo()` メソッドは、iOS 環境および Android 環境の場合に使用可能です。
- コールバックは関数仮パラメーターとして指定する必要があります。

```
function wlCommonInit() {  
    WL.App.hideSplashScreen();  
    WL.Device.getNetworkInfo(getNetworkInfoCallback);  
}  
function getNetworkInfoCallback(info) {  
    alert(JSON.stringify(info));  
}
```

- コールバックは、以下のプロパティを持つオブジェクトを受け取ります。
 - `isAirplaneMode` – true/false
 - `carrierName` – スtring (例えば、AT&T または VERIZON)
 - `telephonyNetworkType` – スtring (例えば、UMTS または GPRS)
 - `isRoaming` – true/false
 - `networkConnectionType` – mobile/WiFi
 - `ipAddress` – スtring
 - `isNetworkConnected` – true/false

アジェンダ

- オフライン・モードでの作業
- アクティブ検出 - メソッドの使用
- パッシブ検出 - オフライン・イベントおよびオンライン・イベント
- 追加のメソッド
- フォアグラウンド・イベント
- Worklight ハートビート
- サンプル

フォアグラウンド・イベント

- Worklight アプリケーションがフォアグラウンドに戻ると、Cordova "resume" イベントが発生します。
- このイベントのリスナーを追加し、リスナーに対応するコールバック関数を指定することができます。

```
document.addEventListener("resume", function() {  
    WL.Device.getNetworkInfo( function(networkInfo) {  
        if (networkInfo.isNetworkConnected) {  
            // Perform client or server actions.  
        }  
    });  
}, false);
```

アジェンダ

- オフライン・モードでの作業
- アクティブ検出 - メソッドの使用
- パッシブ検出 - オフライン・イベントおよびオンライン・イベント
- 追加のメソッド
- フォアグラウンド・イベント
- Worklight ハートビート
- サンプル

Worklight ハートビート

- Worklight ハートビートは、接続を確認するために、指定された間隔でサーバーを ping します。
- ハートビートを使用して、アプリケーションがサーバーに接続されたままであることを定期的に確認することができます。
- `WL.Events.WORKLIGHT_IS_CONNECTED` イベントと `WL.Events.WORKLIGHT_IS_DISCONNECTED` イベントはどちらも、指定されたケースでハートビートによって発生することがあります。
- `WL.Client.setHeartBeatInterval(intervalSeconds)` API を使用してハートビート間隔を指定することができます。
- Worklight ハートビートの使用法は、このトレーニング・モジュールのサンプル・コードで確認することができます。

アジェンダ

- オフライン・モードでの作業
- アクティブ検出 - メソッドの使用
- パッシブ検出 - オフライン・イベントおよびオンライン・イベント
- 追加のメソッド
- フォアグラウンド・イベント
- Worklight ハートビート
- サンプル

定期的な接続テストのサンプル

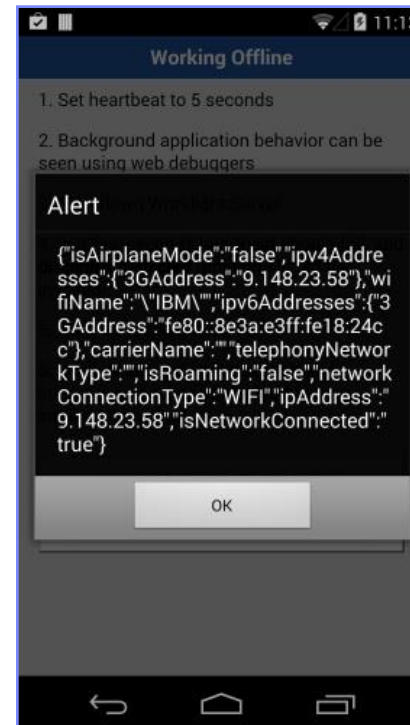
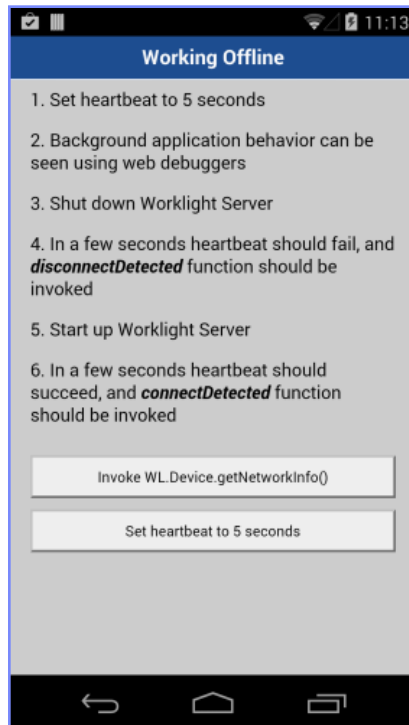
- 次のサンプルは、オフラインおよびオンラインを検出するメカニズムを示しています。

```
document.addEventListener(WL.Events.WORKLIGHT_IS_DISCONNECTED,  
MyApp.connectionFailure, false); ①  
MyApp.connectionFailure = function() {  
  WL.Client.connect({ ②  
    onSuccess: function() {  
      WL.Logger.debug("online");  
      MyApp.onlineRestored(); ③  
    },  
    onFailure: function() {  
      WL.Logger.debug("Still offline... Trying to connect again in 5  
seconds.");  
      window.setTimeout(MyApp.connectionFailure, 5000); ④  
    }  
  });  
};
```

- WL.Events.WORKLIGHT IS DISCONNECTED イベントのイベント・リスナーがドキュメントに追加されます。このイベントの発生時に MyApp.connectionFailure() が呼び出されます。
- WL.Client.connect() がサーバー接続を確立しようとします。
- 接続が正常に確立されると、MyApp.onlineRestored() が呼び出されます。
- 接続が失敗した場合は、MyApp.connectionFailure() を再度呼び出すために 5 秒のタイムアウトが設定されます。

Worklight ハートビートのサンプル

- このトレーニング・モジュールのサンプルは、「IBM Worklight Foundation ユーザー文書」(<https://www.ibm.com/developerworks/mobile/worklight/getting-started/>)の「入門」ページにあります。



このモジュールの題材に関する確認テストを行います。答えは次のスライドにあります。

- アクティブなサーバーのセッションを維持するための最善の方法はどれですか。
 - セッションの有効期限が切れそうになっているため、セッションを維持する何らかのアクションを実行するようユーザーにアラートを出します。
 - バックグラウンドでいずれかの Worklight アダプター・プロシージャーを呼び出して、応答を無視します。
 - ハートビート間隔を、Worklight Server で定義されたセッション・タイムアウトより低い値に設定します。
 - クライアントからセッションを維持する方法はありません。サーバーは、セッションの維持を制御できる唯一のエンティティです。
- Worklight Server に到達できないためにアダプター・プロシージャー呼び出しが失敗した場合、何が起こりますか。
 - クライアント・サイドのフレームワークが、成功するまでプロシージャーを呼び出そうとします。
 - 呼び出しオプションで指定された onFailure コールバックが呼び出されます。開発者は、これを使用してサーバー接続の問題に対処する必要があります。
 - 呼び出しオプションで指定された onFailure コールバックが呼び出されます。さらに、`WL.Events.WORKLIGHT_IS_DISCONNECTED` イベントが発生します。開発者は、これらのいずれかを使用してサーバー接続の問題に対処する必要があります。
 - クライアント・サイドのフレームワークは、「再試行 (retry)」ボタンとともにエラー・メッセージを表示します。
- `WL.Device.getNetworkInfo` API を使用して取得できないのは、どの情報ですか。
 - デバイスの現行 IP アドレス
 - 現在使用中のテレフォニー・タイプ (例えば、GSM または UMTS)
 - デバイスがローミング中であるかどうか
 - デバイスの電話番号

- アクティブなサーバーのセッションを維持するための最善の方法はどれですか。
 - セッションの有効期限が切れそうになっているため、セッションを維持する何らかのアクションを実行するようユーザーにアラートを出します。
 - バックグラウンドでいずれかの Worklight アダプター・プロシージャーを呼び出して、応答を無視します。
 - ハートビート間隔を、Worklight Server で定義されたセッション・タイムアウトより低い値に設定します。
 - クライアントからセッションを維持する方法はありません。サーバーは、セッションの維持を制御できる唯一のエンティティです。
- Worklight Server に到達できないためにアダプター・プロシージャー呼び出しが失敗した場合、何が起こりますか。
 - クライアント・サイドのフレームワークが、成功するまでプロシージャーを呼び出そうとします。
 - 呼び出しオプションで指定された onFailure コールバックが呼び出されます。開発者は、これを使用してサーバー接続の問題に対処する必要があります。
 - 呼び出しオプションで指定された onFailure コールバックが呼び出されます。さらに、`WL.Events.WORKLIGHT IS DISCONNECTED` イベントが発生します。開発者は、これらのいずれかを使用してサーバー接続の問題に対処する必要があります。
 - クライアント・サイドのフレームワークは、「再試行 (retry)」ボタンとともにエラー・メッセージを表示します。
- `WL.Device.getNetworkInfo` API を使用して取得できないのは、どの情報ですか。
 - デバイスの現行 IP アドレス
 - 現在使用中のテレフォニー・タイプ (例えば、GSM または UMTS)
 - デバイスがローミング中であるかどうか
 - デバイスの電話番号

特記事項

- これらの資料は、以下のご使用条件に同意していただける場合に限りご使用いただけます。
- 本書は米国 IBM が提供する製品およびサービスについて作成したものです。
- 本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。
- IBM は、本書に記載されている内容に関して特許権（特許出願中のものを含む）を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。
 - 〒103-8510
東京都中央区日本橋箱崎町19番21号
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外

- 以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。
- この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。
- 本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。
- IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。
- 本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム（本プログラムを含む）との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。
 - IBM Corporation
Dept F6, Bldg 1
294 Route 100
Somers NY 10589-3216
USA

- 本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。
- 本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。
- IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

著作権使用許諾:

- 本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほめめしたり、保証することはできません。
- それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。
 - © (お客様の会社名) (西暦年) このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。© Copyright IBM Corp. 年を入れる。 All rights reserved.

プライバシー・ポリシーの考慮事項

- サービス・ソリューションとしてのソフトウェアも含めた IBM ソフトウェア製品（「ソフトウェア・オファリング」）では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie はじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的な事項を確認ください。
- このソフトウェア・オファリングは、展開される構成に応じて、(アプリケーション・サーバーが生成する) セッション情報を収集するセッションごとの Cookie を使用場合があります。これらの Cookie は個人情報を含まず、セッション管理のために要求されるものです。加えて、匿名ユーザーの認識および管理のために持続的な Cookie が無作為に生成される場合があります。これらの Cookie も個人情報を含まず、要求されるものです。
- この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人を特定できる情報を収集する機能を提供する場合、お客様は、このような情報を収集するにあたって適用される法律、ガイドライン等を遵守する必要があります。これには、エンドユーザーへの通知や同意の要求も含まれますがそれらには限られません。このような目的での Cookie を含む様々なテクノロジーの使用の詳細については、IBM の『IBM オンラインでのプライバシー・ステートメント』(<http://www.ibm.com/privacy/details/jp/ja/>) の『クッキー、ウェブ・ビーコン、その他のテクノロジー』および『IBM Software Products and Software-as-a-Service Privacy Statement』(<http://www.ibm.com/software/info/product-privacy>) を参照してください。

サポートおよびコメント

- IBM Worklight の一連の文書、トレーニング資料、および質問をポストできるオンライン・フォーラムはすべて、次の IBM Web サイトからご覧になれます。
 - <http://www.ibm.com/mobile-docs>
- サポート
 - ソフトウェア・サブスクリプション & サポート (ソフトウェア・メンテナンスと呼ばれる場合もあります) は、パスポート・アドバンテージおよびパスポート・アドバンテージ・エクスプレスから購入されたライセンスに含まれています。International Passport Advantage Agreement および IBM International Passport Advantage Express Agreement の追加情報については、次のパスポート・アドバンテージ Web サイトを参照してください。
 - <http://www.ibm.com/software/passportadvantage>
 - ソフトウェア・サブスクリプション & サポートが有効になっている場合、IBM は、インストールおよび使用法 (ハウツー) に関する短期間の FAQ に対するサポートや、コード関連の質問に対するサポートを提供します。詳しくは、次の IBM ソフトウェア・サポート・ハンドブックを参照してください。
 - <http://www.ibm.com/support/handbook>
- ご意見
 - 本資料に関するご意見をお寄せください。本資料の具体的な誤りや欠落、正確性、編成、題材、または完成度に関するご意見をお寄せください。お寄せいただくご意見は、本マニュアルまたは製品の情報、およびその情報の提示方法に関するもののみとしてください。
 - 製品の技術的な質問および情報、および価格については、担当の IBM 営業所、IBM ビジネス・パートナー、または認定リマーカーターにお問い合わせください。
 - IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。IBM またはいかなる組織も、お客様から提示された問題についてご連絡を差し上げる場合にのみ、お客様が提供する個人情報を使用するものとします。
 - どうぞよろしくお願いいたします。
 - 次の IBM Worklight Developer Edition サポート・コミュニティにご意見をお寄せください。
 - <https://www.ibm.com/developerworks/mobile/worklight/connect.html>
 - IBM からの回答を希望される場合は、以下の情報をご連絡ください。
 - 氏名
 - 住所
 - 企業または組織
 - 電話番号
 - Eメール・アドレス

ありがとうございました

