

マルチデバイス対応の モバイル&クラウド連携アプリ 超高速開発！

- Xamarin, Visual Studio, Microsoft Azure Mobile Services -

鈴木 章太郎

テクニカルエバンジェリスト

日本マイクロソフト株式会社

デベロッパーエクスペリエンス & エバンジェリズム統括本部

<http://blogs.msdn.com/b/shosuz/>

- テクニカルエバンジェリスト
<http://blogs.msdn.com/b/shosuz>
- MTC アーキテクト
<http://www.microsoft.com/ja-jp/business/mtc/ads.aspx>
- 中央大学 総合政策学部 非常勤講師
- ASPIC 執行役員
- Microsoft 軽音楽部 広報担当 (Guitar/Keyboard)
- 吹きネタは主に、Windows、Windows Azure、Windows Phone, RIA, HTML5, MVVM, iOS/Android x Microsoft Azure 連携, Guitar ... 等



本セッションの目的

iOS / Android / Windows

すべてに対応したクラウド連携アプリを

 **Visual Studio** と **Xamarin**

WINDOWS AZURE

MOBILE SERVICES とで

で超高速に開発する方法をご理解いただく

最近の開発事情

スマートデバイス向けの
アプリ開発で求められていること

すばや
い
市場投入



コストを
抑える



質の高い
UX



進化、多様化するプラットフォーム

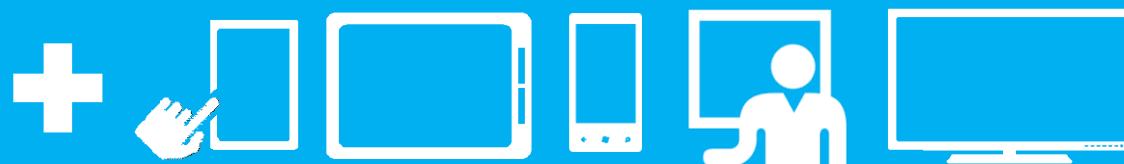
デバイス & サービス 両方に クロスプラットフォーム対応の必要性

プラットフォーム

クライアント



デバイス



サーバー



サービス



デバイス&サービス、これからの超高速開発

デバイス

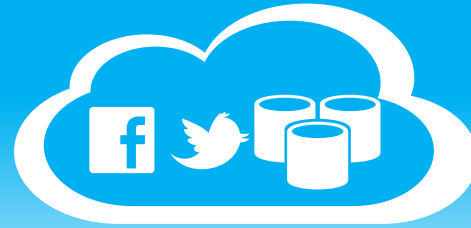


Xamarin



デバイス アプリ

サービス



Microsoft Azure



モバイル サービス

社内システム



サーバ & クライアント

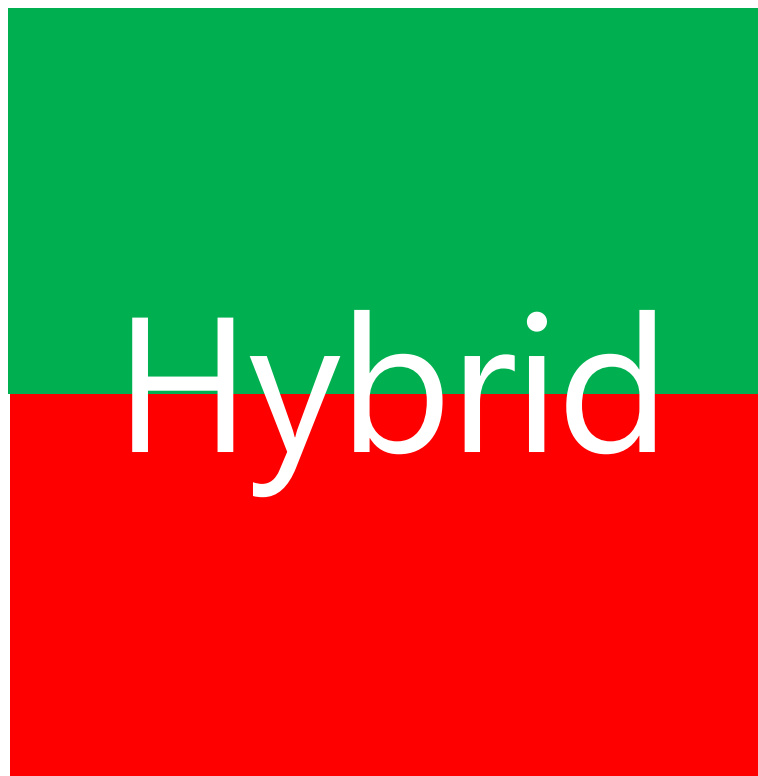
既存の基幹システム
パッケージ活用



Visual Studio & Team Foundation Server / Visual Studio Online



クロスプラットフォーム開発



OS (iOS, Android, Windows)

Xamarin とは？ その特長

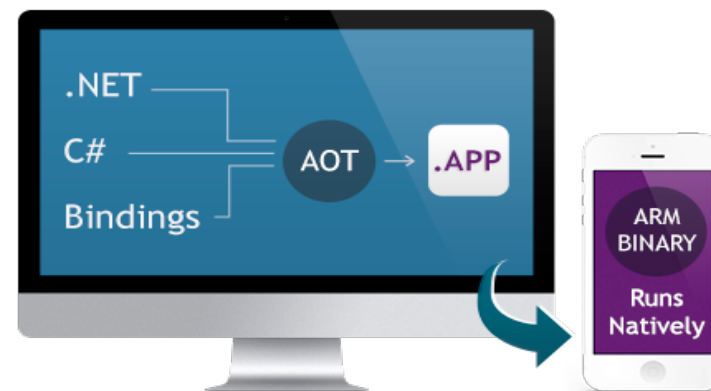


Xamarin  Visual Studio

“フル” ネイティブ
すべての API が使える

高パフォーマンス
コンパイルされたパッケージ

C#, .NET, Visual Studio
既存の資産やスキルを活用



Xamarin の仕組み



ネイティブ

Storyboard/ XIB

ユーザインターフェース
機能の具体化
依存コード

参照

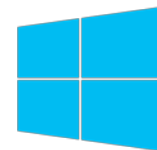


ネイティブ

AXML

ユーザインターフェース
機能の具体化
依存コード

参照



ネイティブ

XAML

ユーザインターフェース
機能の具体化
依存コード

参照

ロジック・データ・抽象化された機能 - C# & .NET



Visual Studio

Xamarin

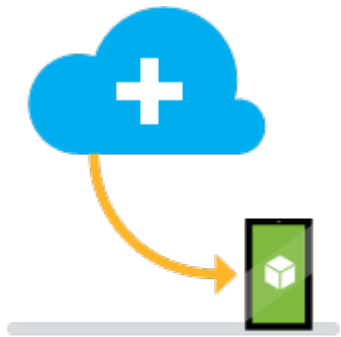
Xamarin による マルチデバイス開発 DEMO

MBaaS : Mobile Backend as a Services

バックエンドの共通化

モバイルアプリに迅速にクラウドバックエンドを追加

データ
ベース



認証



プッシュ
通知



スケール



マルチ
デバイス



最新 Microsoft Azure Mobile Services

クライアント SDK

Windows ストア

iOS

Android

Xamarin

PhoneGap

Sencha



Windows Phone

iOS

Android

HTML 5/JS



REST API

データベース



SQL



Table
Storage



Blob
Storage



Mongo
DB



Hybrid
Connections

ユーザー認証



Facebook



Twitter



Microsoft



Google



Azure
Active
Directory

プッシュ通知



WNS &
MPNS



APNS



GCM



Notification
Hubs

バック エンド



Node.js
Express

ASP.NET
Web API



ソース
管理

Microsoft Azure Mobile Services による高速なバックエンド開発 DEMO

まとめ



- マルチデバイス対応アプリを超高速に開発し、クラウドへの展開が可能
- Visual Studio + Xamarin によるマルチデバイスアプリ開発が可能
- Microsoft Azure Mobile Services によるバックエンドの高速な開発



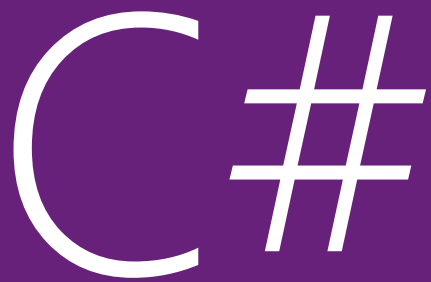
© 2014 Microsoft Corporation. All rights reserved.

This presentation is for informational purposes only. Microsoft makes no warranties, express or implied, in this summary.

このプレゼンテーションは、情報提供のみを目的としています。Microsoft は、この概要について、明示または暗示を問わず、いかなる保証も行いません。

參考資料

Appendix



**Visual Studio +
Xamarin による
マルチデバイスアプリ開発**

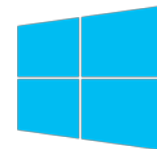
Xamarin の仕組み



ネイティブ
Storyboard/ XIB



ネイティブ
AXML



ネイティブ
XAML

ビジネス ロジック C# & .NET



Visual Studio

Xamarin

Windows ストア アプリ



ネイティブ
XAML

ビジネス ロジック
C#

Visual Studio with
Xamarin

思っているよりも簡単
世界中の Windows ユーザにリーチ



Android アプリ



ネイティブ
AXML

ビジネス ロジック
C#

Visual Studio with
Xamarin

Visual Studioで完結
画面デザイナーも使える



iOS アプリ



ネイティブ
Storyboard / Xib

ビジネス ロジック
C#

Visual Studio with
Xamarin

C# でコーディング
ユーザインタフェースはネイティブ



開発環境

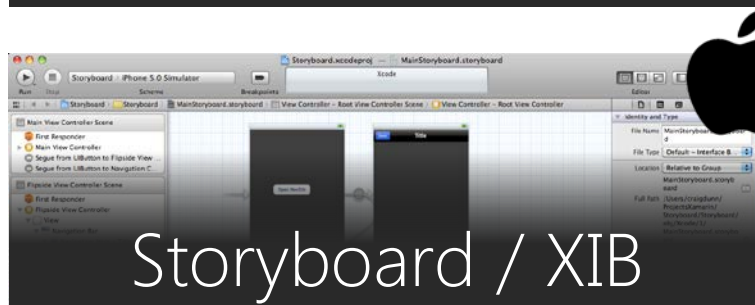
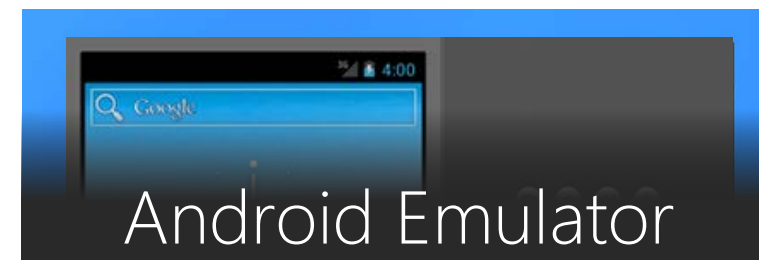
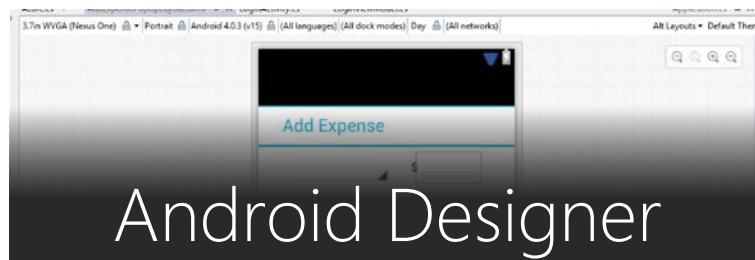
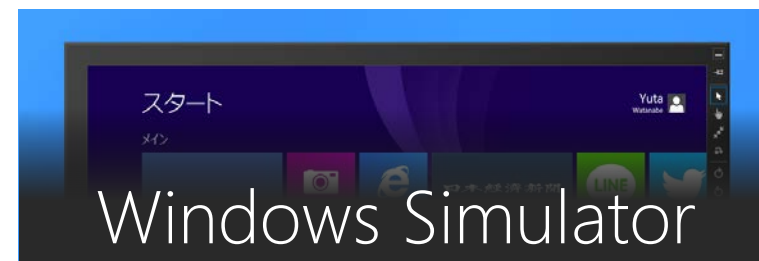
ビジネス ロジック



画面デザイン



デバッグ



PCL (Portable Class Library)とは



ネイティブ

ビジネス ロジック
C#

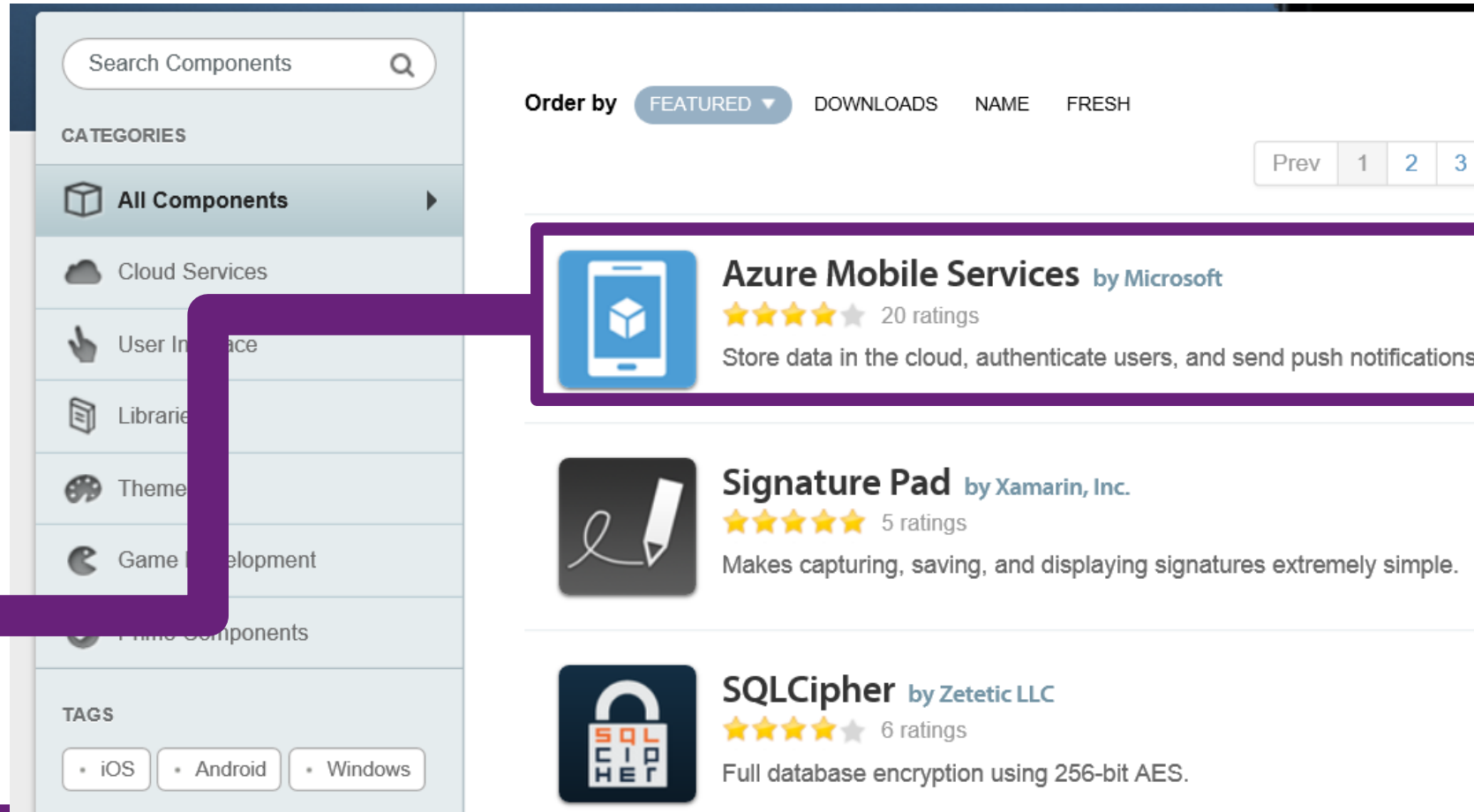
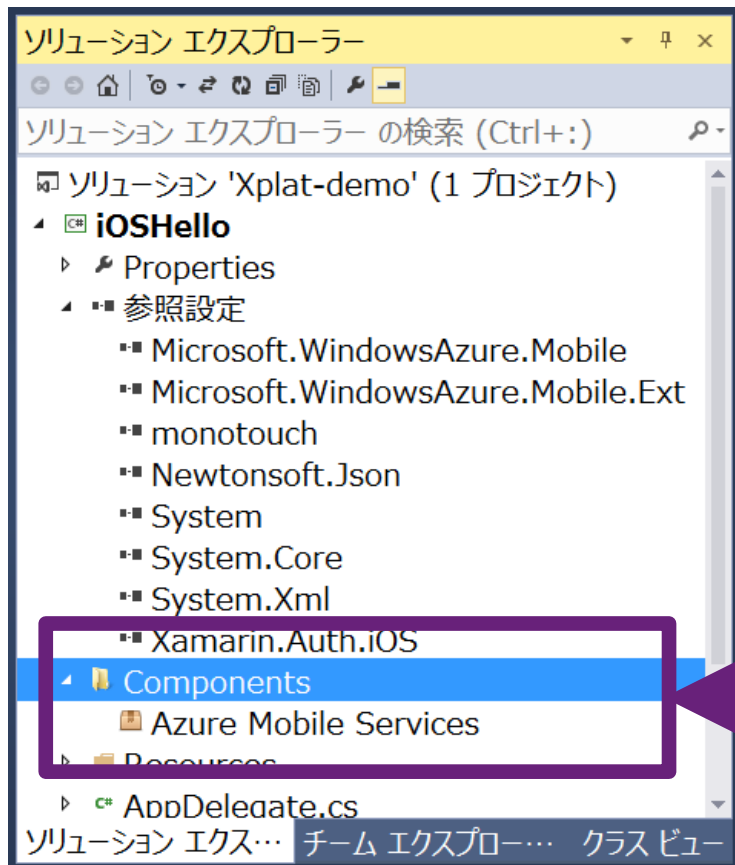
Visual Studio with
Xamarin

複数の .NET 処理系に対して、
バイナリレベルで再利用できる
ライブラリ

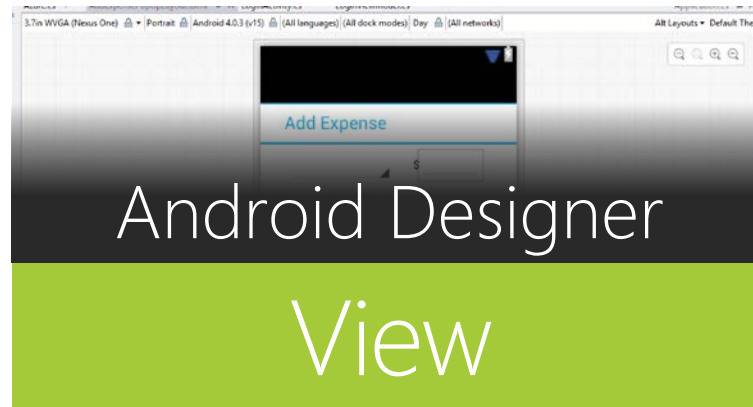
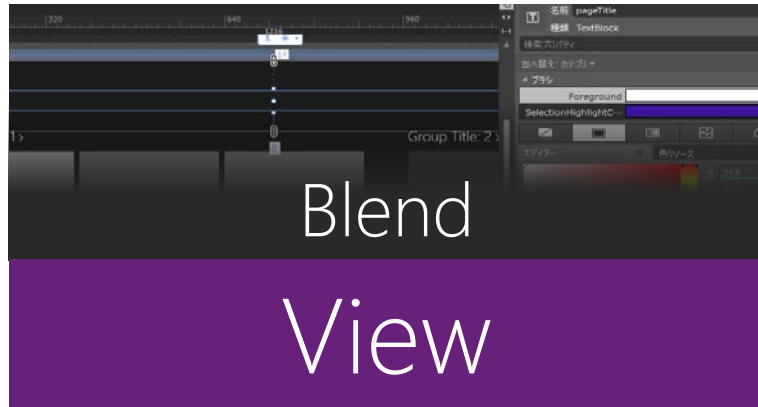


Xamarin Component Store

追加コンポーネントで簡単機能追加



MvvmCross



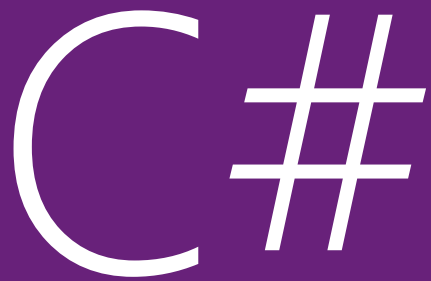
ViewModel

Model

PCL(Portable Class Library)

Premier Support for Developers





Microsoft Azure Mobile Services による マルチデバイスアプリ開発

Mobile Services が提供するサービス



- データアクセス、プッシュ通知、認証サポート

Windows 8、Windows Phone 8 SDK、iOS SDK、Android SDK、HTML5 SDK



- 強力なサーバーサイドスクリプトのサポート

プッシュ通知、バリデーション、プリプロセッシング、ポストプロセッシング、他の Web サービスとの連携

Node.js で様々な高度なクエリーが実行可能

posts : 読み取り (JavaScript)

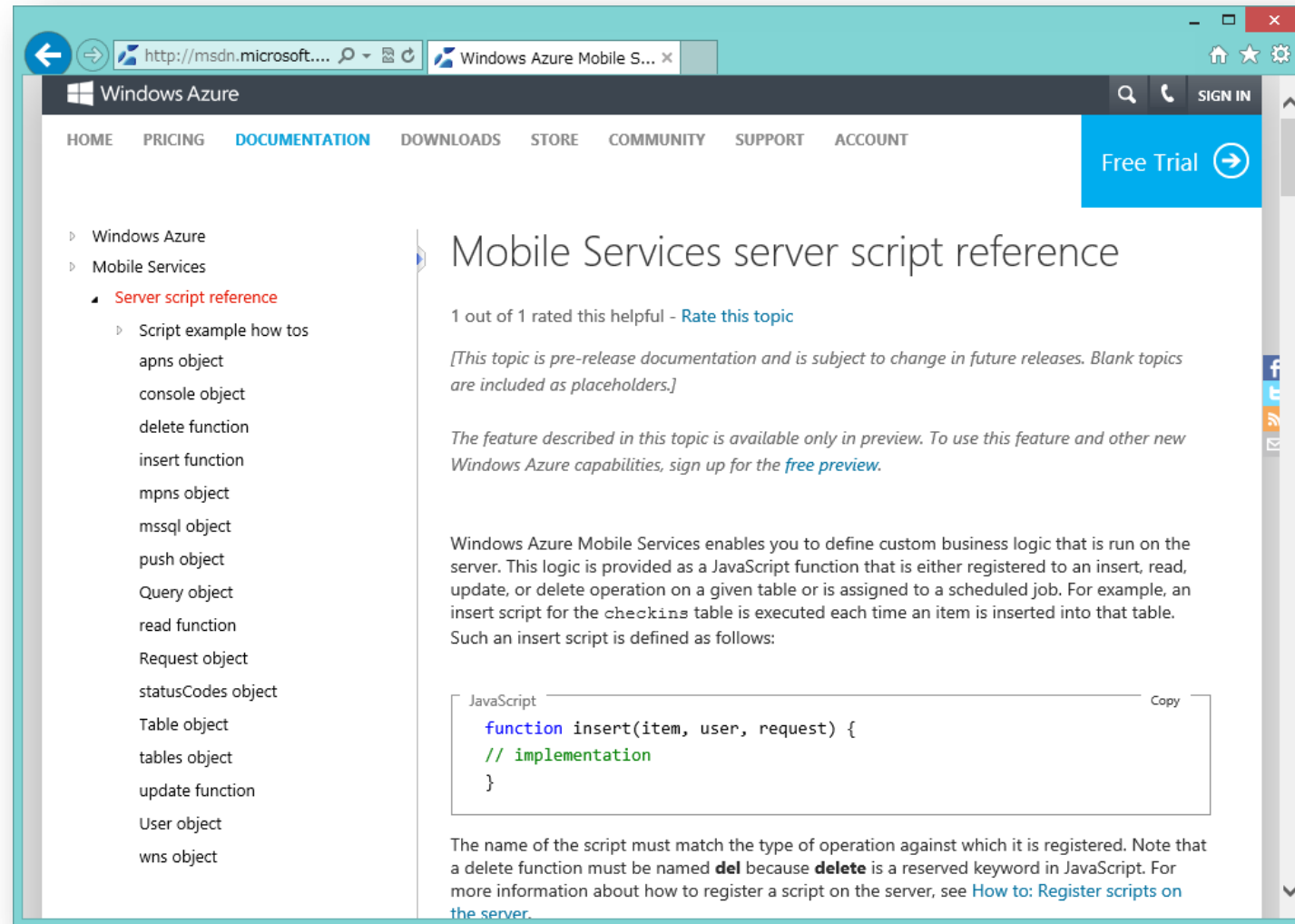
```
function read(query, user, request) {
  query.orderByDescending('id');
  request.execute({ success:
function(results) {
  if (results.length === 0) {
    request.respond();
    return;
  }

  var postIds = [];
  results.forEach(function(p){
    postIds.push(p.id);
  });
```

```
var sql=
  "SELECT posttags.postId, tags. id,
      tags.name
  FROM posttags,tags
  WHERE posttags.postId IN
  (";
      sql += postIds.join(",");
      sql += ")
  AND posttags.tagId = tags.id";

console.log(sql);
.....
```

サーバースクリプトリファレンス

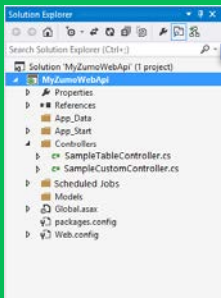


<http://msdn.microsoft.com/en-us/library/windowsazure/jj554226.aspx>

ASP.NET Web API バックエンド

- データモデル (EF-CodeFirst) 、スケジュールドジョブ
- Visual Studio のツールを最大限活用
- ローカルとリモートでデバッグ、発行

ソースコード



Commit hook:
ビルドプロジェクト



Web 発行



Mobile Services 互換 Web
API コントローラー

Website
XDRIVE¥site¥wwwroot



Mobile Services
compatible WebAPI
controllers



Web.config

C:¥...¥MobileServices



Mobile Services
runtime



Web.config

website root

Load



ユーザーDB:

EF CodeFirst
またはカスタム
マイグレーション

埋め込まれた
App settings

オフラインデータ操作

エンタープライズモバイルアプリの データ利用シナリオ

- ・ ユーザー入力/更新
- ・ ワークフロー
- ・ ローカルテレメトリデータ
- ・ 分散数値計算処理、等々...



データベース作成とテーブルの定義

データベース の作成

- **MobileServiceSQLiteStore**
- データベース作成と
テーブル作成を wrap

テーブルの 作成

- Client /Server 双方に作成
 - オフラインデータ用クラス
- 標準的な SQL DDL は不要

ソーシャル ID 認証



iOS

<http://www.windowsazure.com/ja-jp/develop/mobile/tutorials/get-started-with-users-ios/>

Android

<http://www.windowsazure.com/ja-jp/develop/mobile/tutorials/get-started-with-users-android/>

- Microsoft Account、Twitter、Facebook、Google ID 等
- 各 CRUD 操作のテーブルレベルのアクセス許可
 - ・すべてのユーザー
 - ・アプリケーション キーを持つユーザー
 - ・認証されたユーザーのみ
 - ・スクリプトと管理者のみ
- サーバー側スクリプトを使用したよりきめ細かい制御

HTML5

<http://www.windowsazure.com/ja-jp/develop/mobile/tutorials/get-started-with-users-html/>

Windows 8.1 ストアアプリの場合

Live Connect ポータル

<https://account.live.com/developers/applications/index>

Windows ストア アプリの開発者センター

ホーム | **ダッシュボード** | ドキュメント | Code Recipe | ダウンロード | サポート | コミュニティ

アプリの提出

アプリの名前
アプリには一意の名前を付けてください。
詳細情報

販売の詳しい情報
アプリの価格、カテゴリ、販売する国/地域を選びます。
詳細情報

高度な機能
プッシュ通知と Live サービスを構成してアプリ内販売を定義します。
詳細情報

年齢区分と評価証明書
アプリの対象者について説明し、評価証明書をアップロードします。
詳細情報

暗号化

Live Connect デベロッパー センター

ホーム | **マイ アプリ** | ドキュメント | Interactive SDK | ダウンロード | サポート | ショーケース

マイ アプリケーション

アプリケーションの作成

Windows ストア アプリを登録する場合、このサイトは正しい登録先ではありません。Windows ストアのダッシュボードに移動してください。

Live Connect サービスの状況を表示

	日次ユーザー数	月次ユーザー数
dato 000000004C0C319 (Windows Developer Preview Application)	0	0
MyFirstProject 000000004B01675B	0	0
Inv11 0000000040004D0A	0	0
Virtuoso - Shotaro Suzuki's Blog 000000004C0CB225	0	0
Win8toDo 000000004C0D8679	0	0
win8todolist 00000000440D4A84 (Windows Developer Preview Application)	0	2

次のステップ

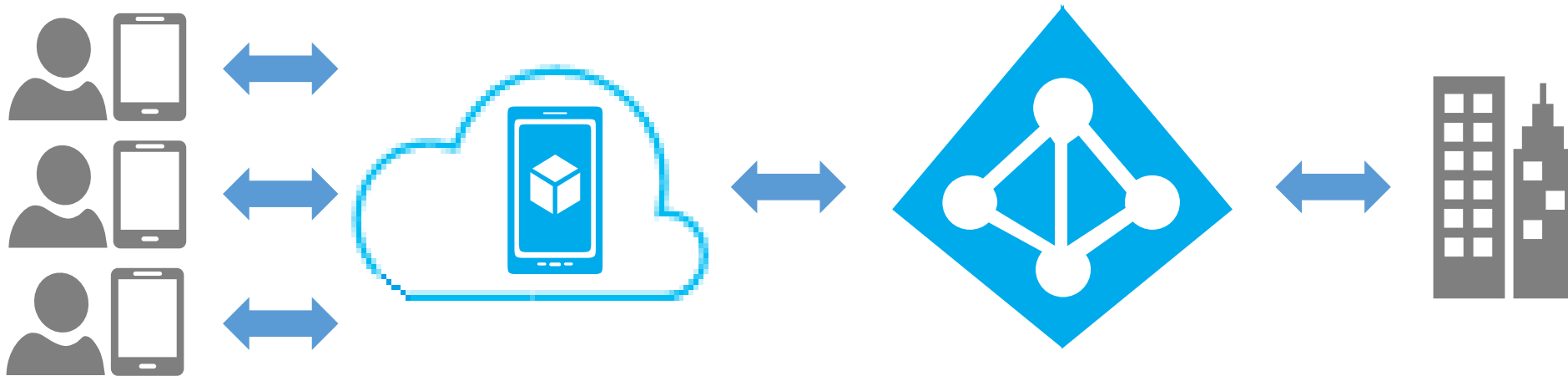
Interactive SDK Web サイトにアクセスして、さまざまな API を確認します。コーディングは必要ありません。
詳細情報

Windows ストア アプリポータル

<http://msdn.microsoft.com/ja-JP/windows/apps>

Azure Active Directory と Mobile Services

- 組織 ID 利用モバイル LOB アプリ開発可能
- Microsoft Azure Active Directory ユーザー



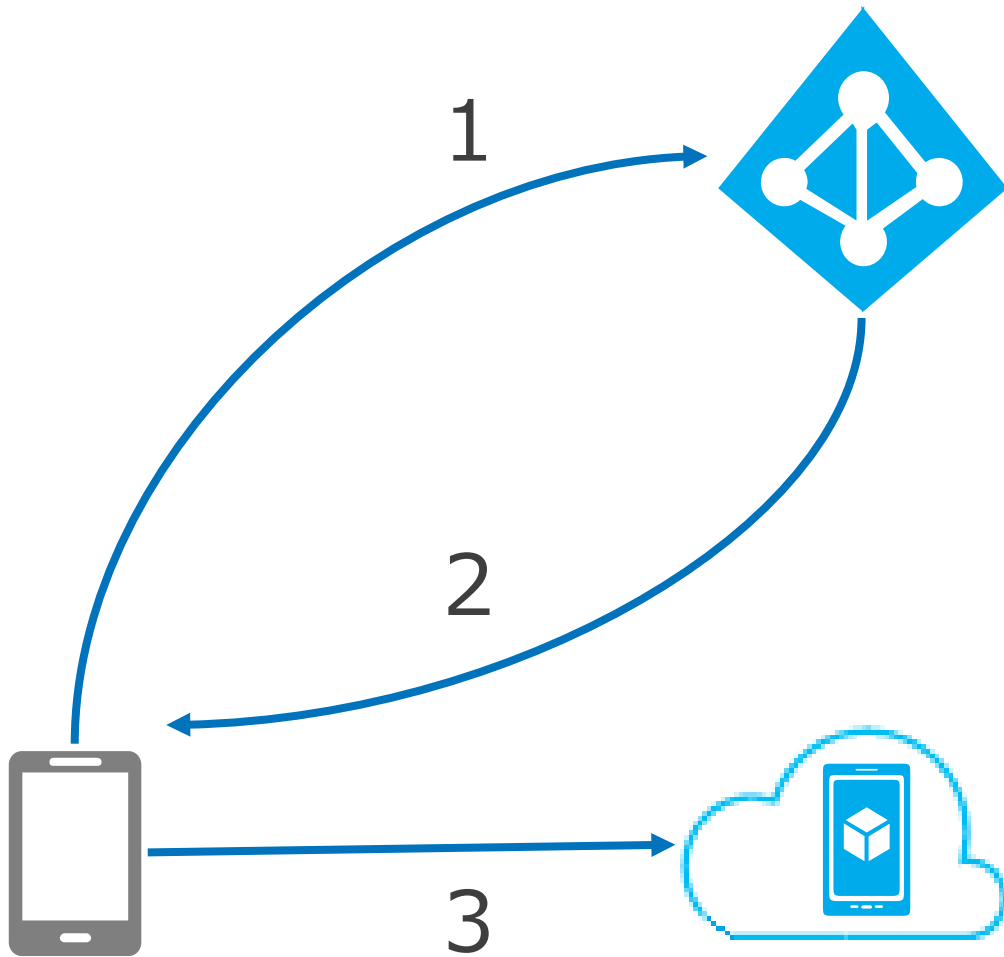
Active Directory Authentication Library (ADAL)

Microsoft Azure Active Directory によって
保護されたリソースへのログインを可能に

複数の企業リソースへの
シングルサインオンを提供

Windows ストアアプリ、iOS/Android アプリ
各プラットフォームで利用可能

基本的な ADAL + Mobile Services のフロー



- 1) クライアントアプリが ADAL を使ってログインを試行。ユーザーが入力したクレデンシャル情報を Azure AD に送信
- 2) Azure AD は Mobile Services のためのアクセストークンと リフレッシュトークンのペアを ADAL に返送
- 3) クライアントは、当該 アクセストークンを Mobile Service に 渡し、Mobile Services トークンに変換して、セッションを継続

基本的な ADAL + Mobile Services のフロー

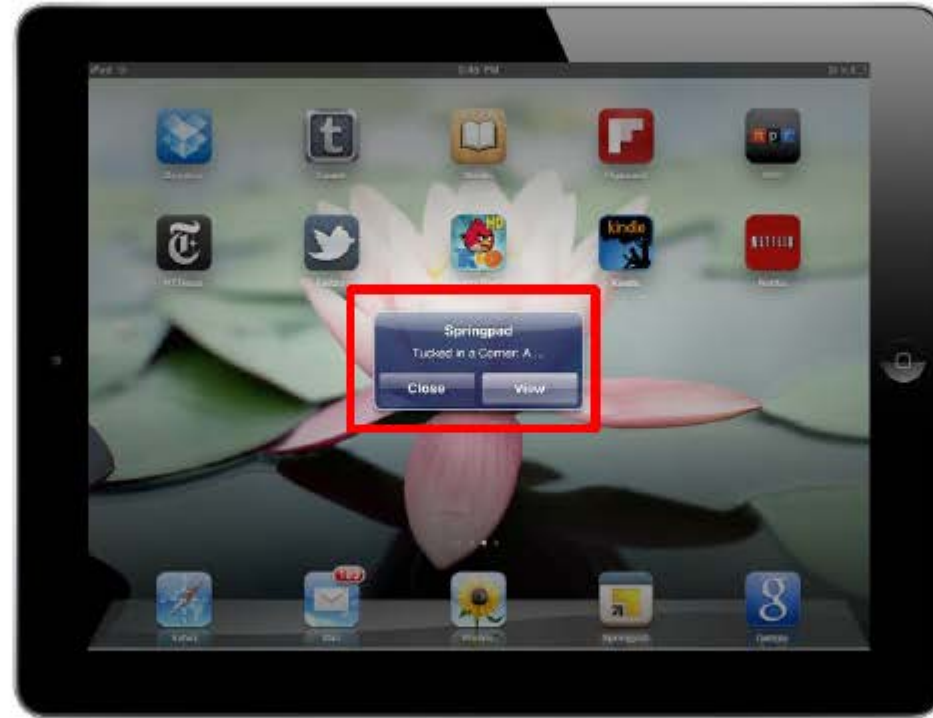
```
string authority = "https://login.windows.net/<テナント名>.onmicrosoft.com";  
string resourceURI = "https://<サービス名>.azure-mobile.net/login/aad";  
string clientID = "<Azure Active Directory ポータルから入手した Client App ID>"
```

```
AuthenticationContext ac = new AuthenticationContext(authority);  
AuthenticationResult ar = await ac.AcquireTokenAsync(resourceURI, clientID);  
string accessToken = ar.AccessToken;
```

// Mobile Services にアクセストークンを付与

```
JObject payload = new JObject();  
payload["access_token"] = accessToken;  
MobileServiceUser user = await  
    App.MobileService.LoginAsync(  
        MobileServiceAuthenticationProvider.  
            WindowsAzureActiveDirectory, payload);
```

プッシュ通知



スタート

Hello!



メール



地図



SkyDrive



Bing 検索



メッセージング

TOPIX
1,088.17 ▲ +0.88% (+9.51)
2013/06/26 10:08 JST



People



デスクトップ



天気



錦織が2回戦ヘ＝クルム伊達も快勝－ウィンブルドンテニス



ゲーム



カメラ



カレンダー



Internet Explorer



ストア

3



IOC評価報告書、東京は「安全、安心、確実」



ミュージック



ビデオ

L

Lync 2013

W

Word 2013

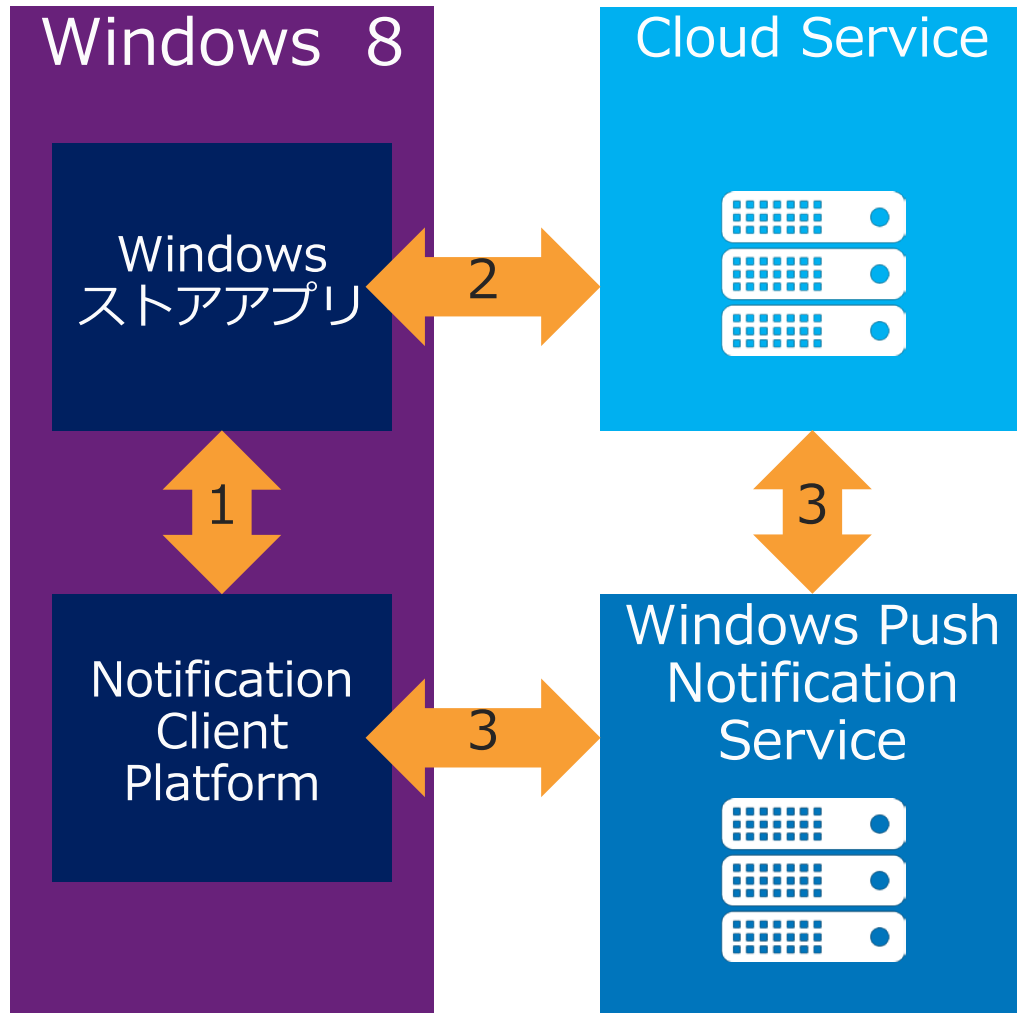
N

OneNote

P

Publisher

Windows ストアアプリ プッシュ通知の仕組み



1. チャンネル URI を要求
2. クラウド サービスに登録
3. 認証とプッシュ通知

iOS 版プッシュ通知の仕組み

- サーバーサイドスクリプトから通知
APNS (Apple Push Notification Services) に依頼



APNSと連携したプッシュ通知

<http://www.windowsazure.com/ja-jp/develop/mobile/tutorials/get-started-with-push-ios/>

Android 版プッシュ通知の仕組み

- サーバーサイドスクリプトから通知
Google Cloud Messaging に依頼



Google Cloud Messaging と連携したプッシュ通知

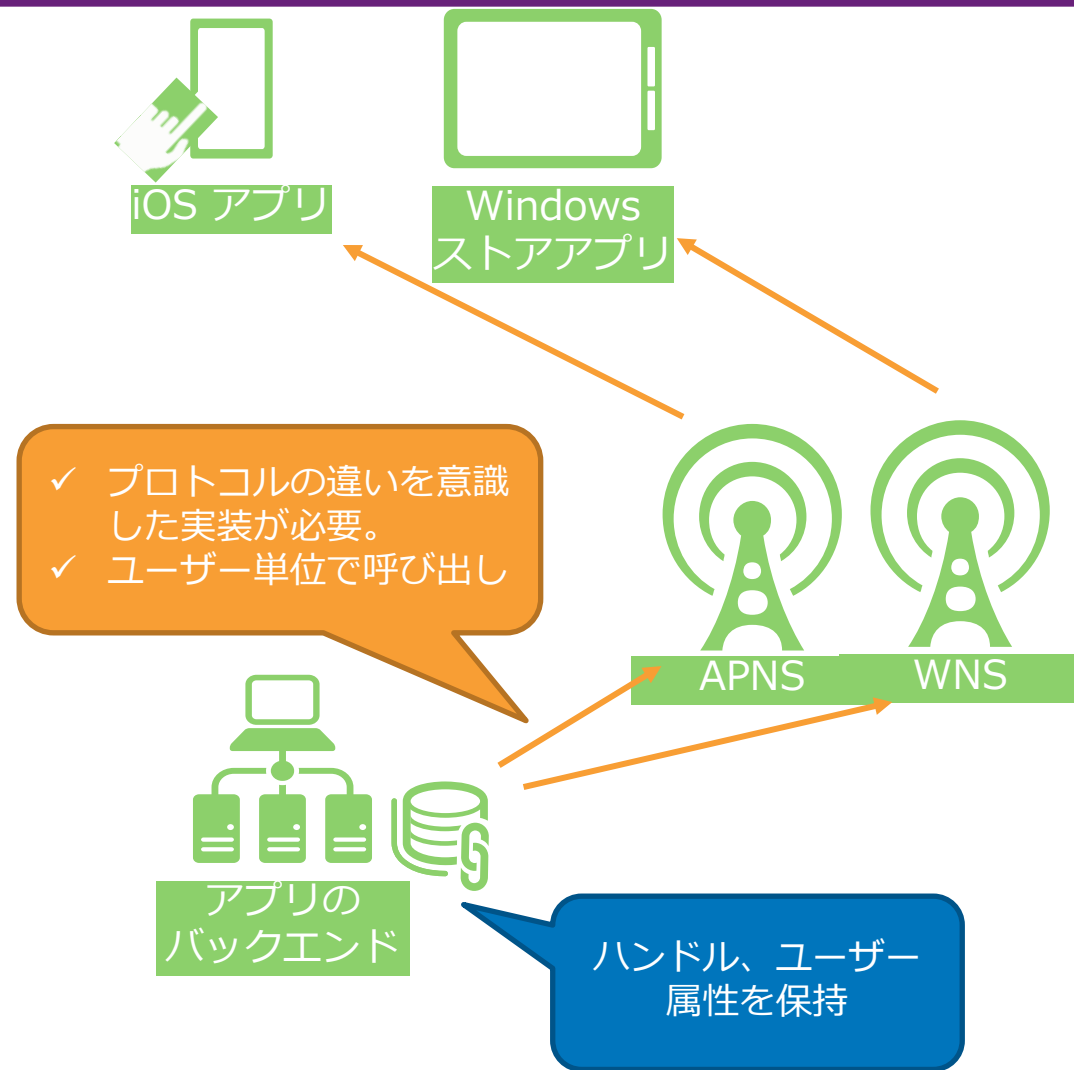
<http://www.windowsazure.com/ja-jp/develop/mobile/tutorials/get-started-with-push-android/>

技術的課題

プラットフォーム毎に仕様が異なる
(iOS、Windows、Androidなど) ため
個別に実装が必要

数千数万、数百万のデバイスに
プッシュ通知するための
スケーラブルなシステム構築は
技術的に難しい

ハンドル、ユーザー属性などの
管理が大変



Microsoft Azure 通知ハブとは？

マルチプラットフォーム対応

(Windows, iOS, Android)

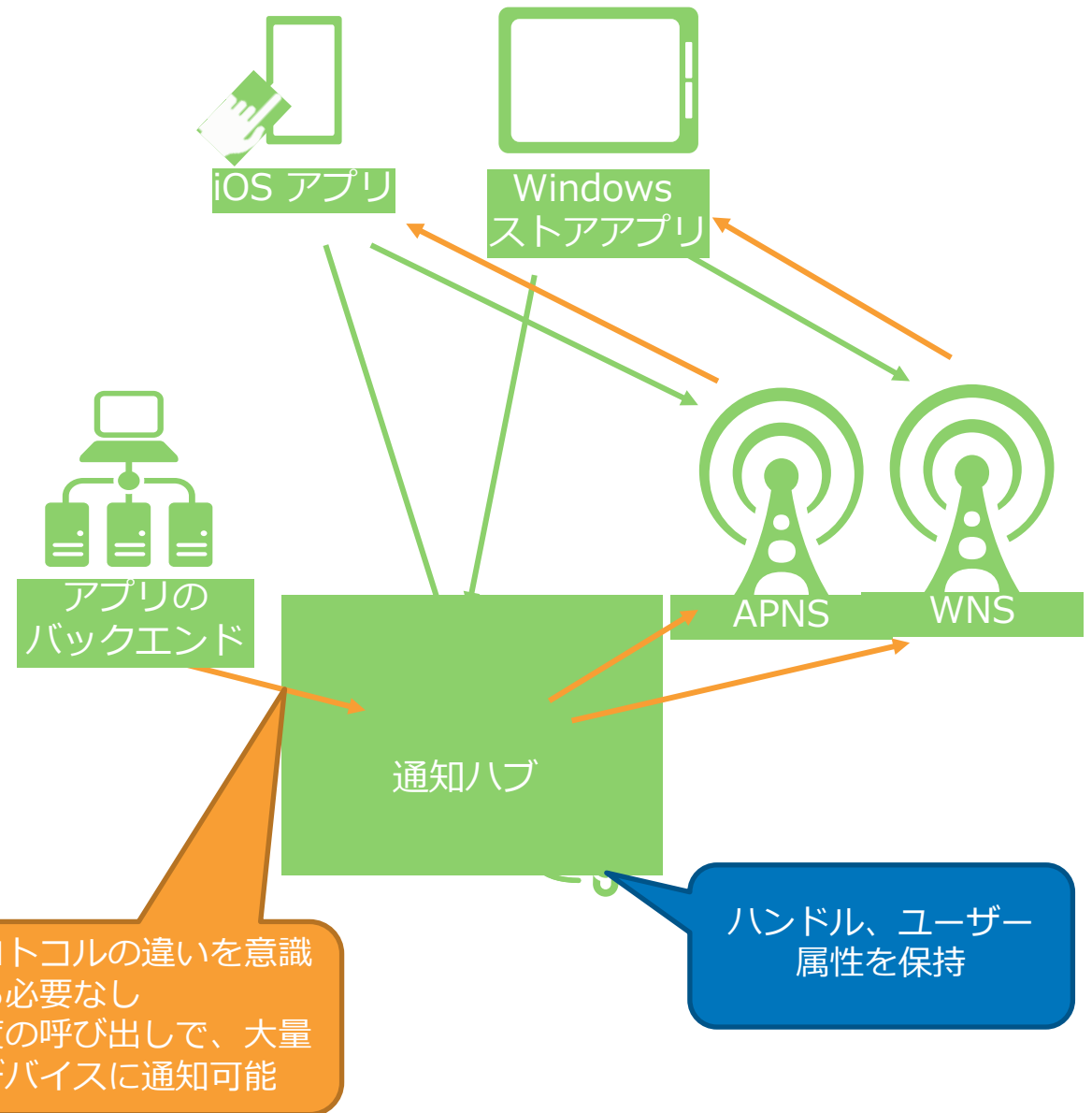
一斉送信やタグベースのフィルタリングも
可能

数千、数万、数百万のデバイスに

低い遅延性で送信できる

スケーラビリティ（例えば地震速報）

ハンドル、ユーザー属性などの管理機能搭
載している



プッシュ通知の活用

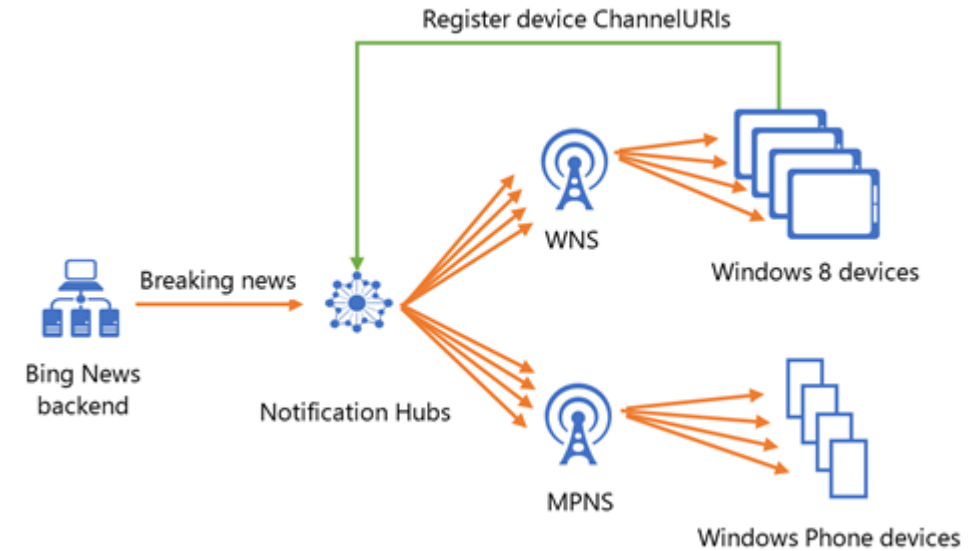
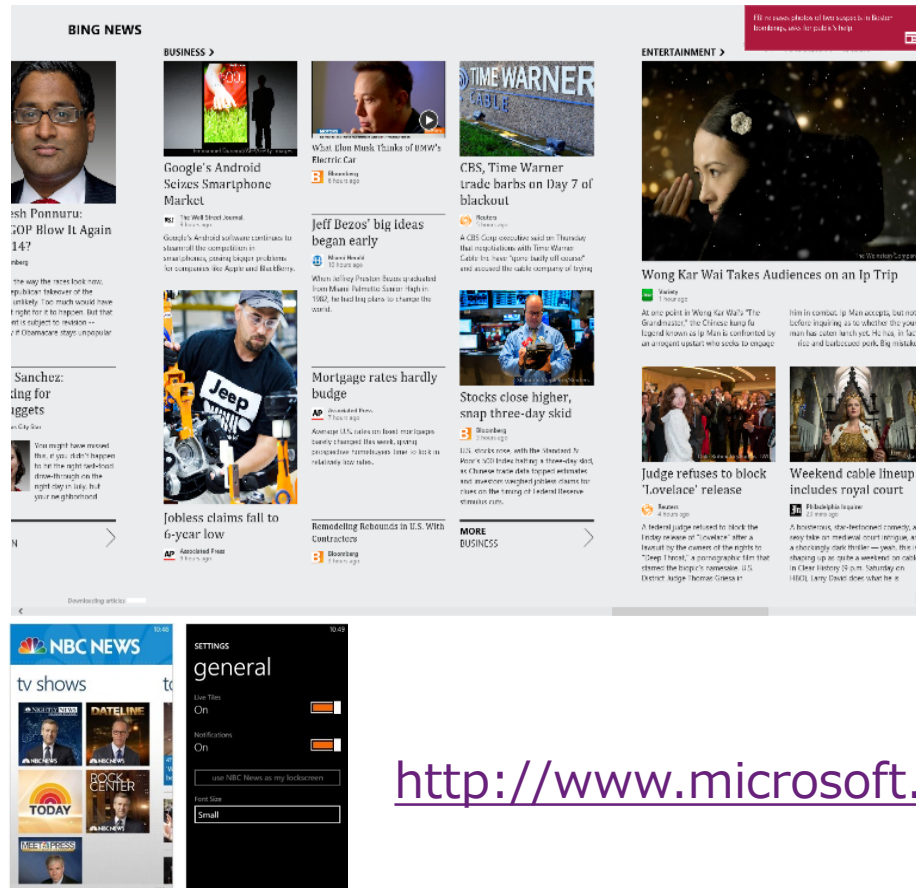
コンシューマーアプリ

- ニュース速報、リマインダー、興味ある新しいアイテム
- SMS よりも安価でパワフル
- アクティブ率を50 - 100% 向上、Eメールと比較して10倍の効果
- ユーザーとのエンゲージメントの強化

コマーシャルアプリ

- BIレポートの生成などのタスク完了時
- システム障害が発生した時
- 予定より長くかかっているスケジュールタスク
- 社員のエクスペリエンスを向上

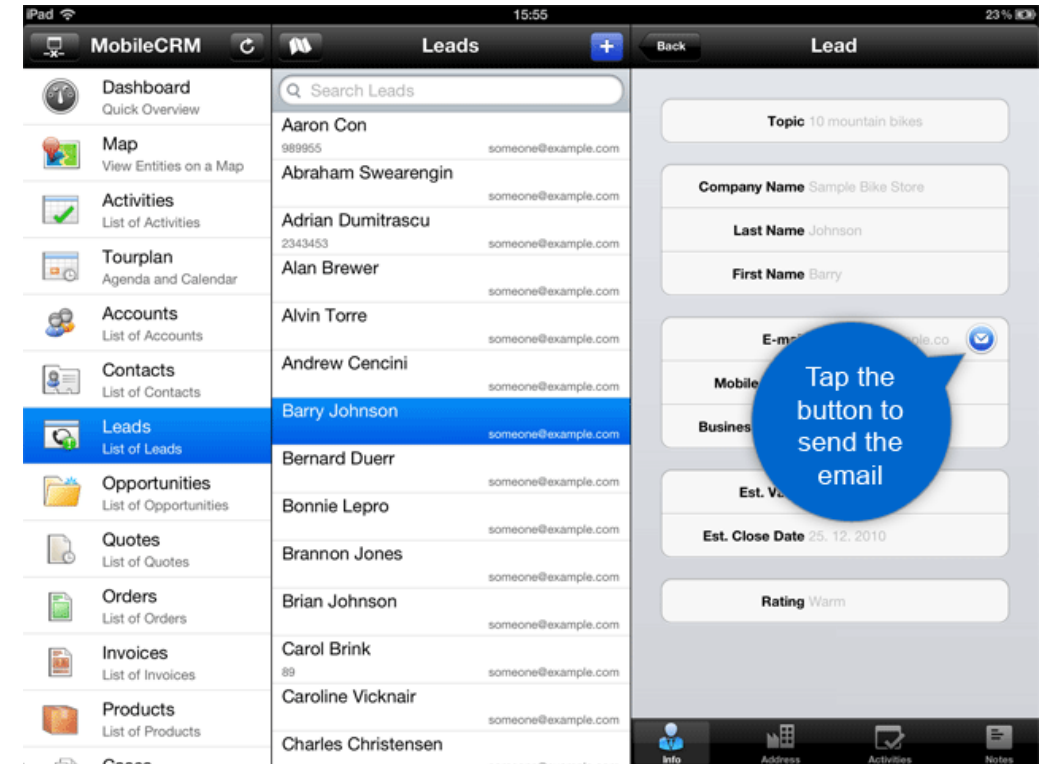
シナリオ：ニュース速報



http://www.microsoft.com/casestudies/Case_Study_Detail.aspx?CaseStudyID=7100

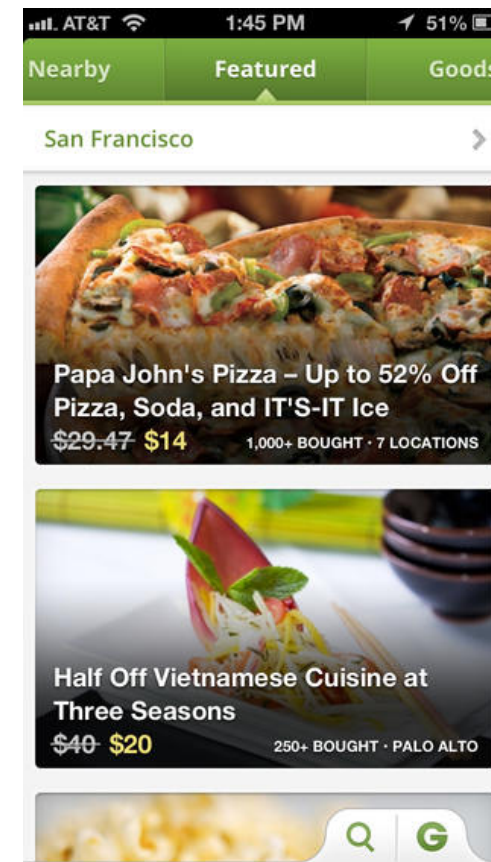
シナリオ: CRM システム

- CRM システムにおけるプッシュ通知
 - アカウントステータスの更新
 - 新しいリードの通知
- 要件
 - マルチクライアントプラットフォーム (Windows/iOS/Android/...)
 - セキュリティ標準
 - マルチ言語対応



シナリオ: ショッピングアプリ

- ショッピングアプリの通知
 - 購入ステータスの通知
 - 購入を完了しなかったユーザーの再ターゲティング
 - クーポンなどのオファーを宣伝
- 通知システムの要件
 - グループینگ
 - フィルタリング
 - セキュリティ



ブランドアプリ

- ブランドアプリの通知
 - アプリのエクスペリエンスの向上 (e.g. ショップを閉店する時の通知する)
 - 振舞やロケーションをベースとして特定のオファー・クーポンのターゲット



© iPhoneSavior.com