

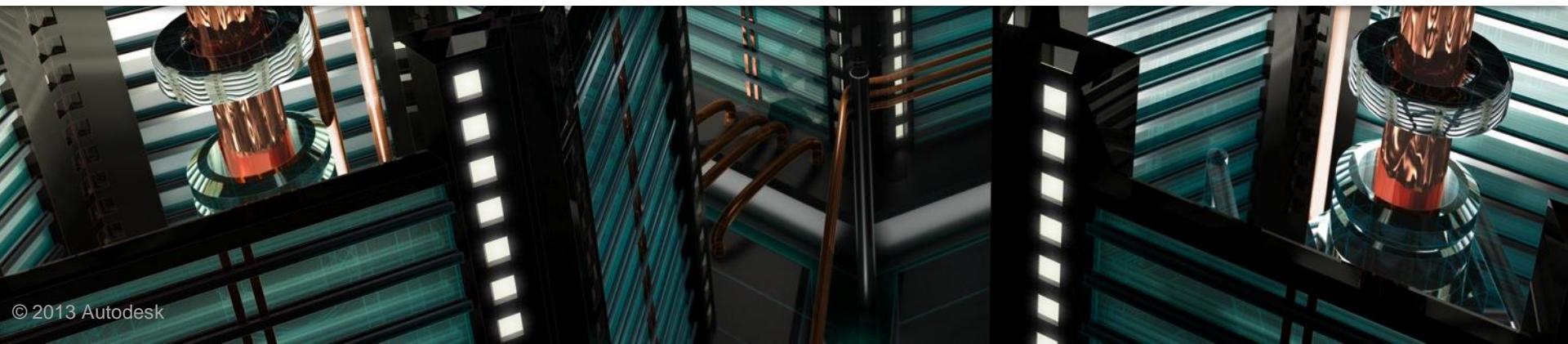


クラウド開発 & Exchange Apps 公開セミナー

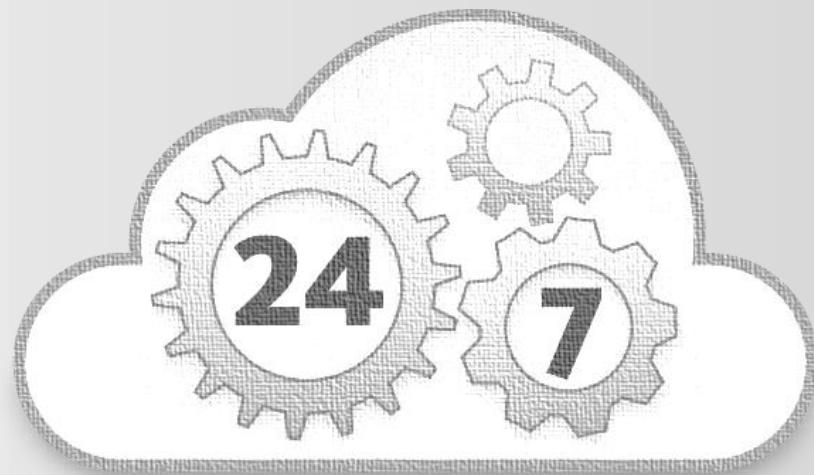
# クラウドを使ったアプリケーション例

伊勢崎 俊明

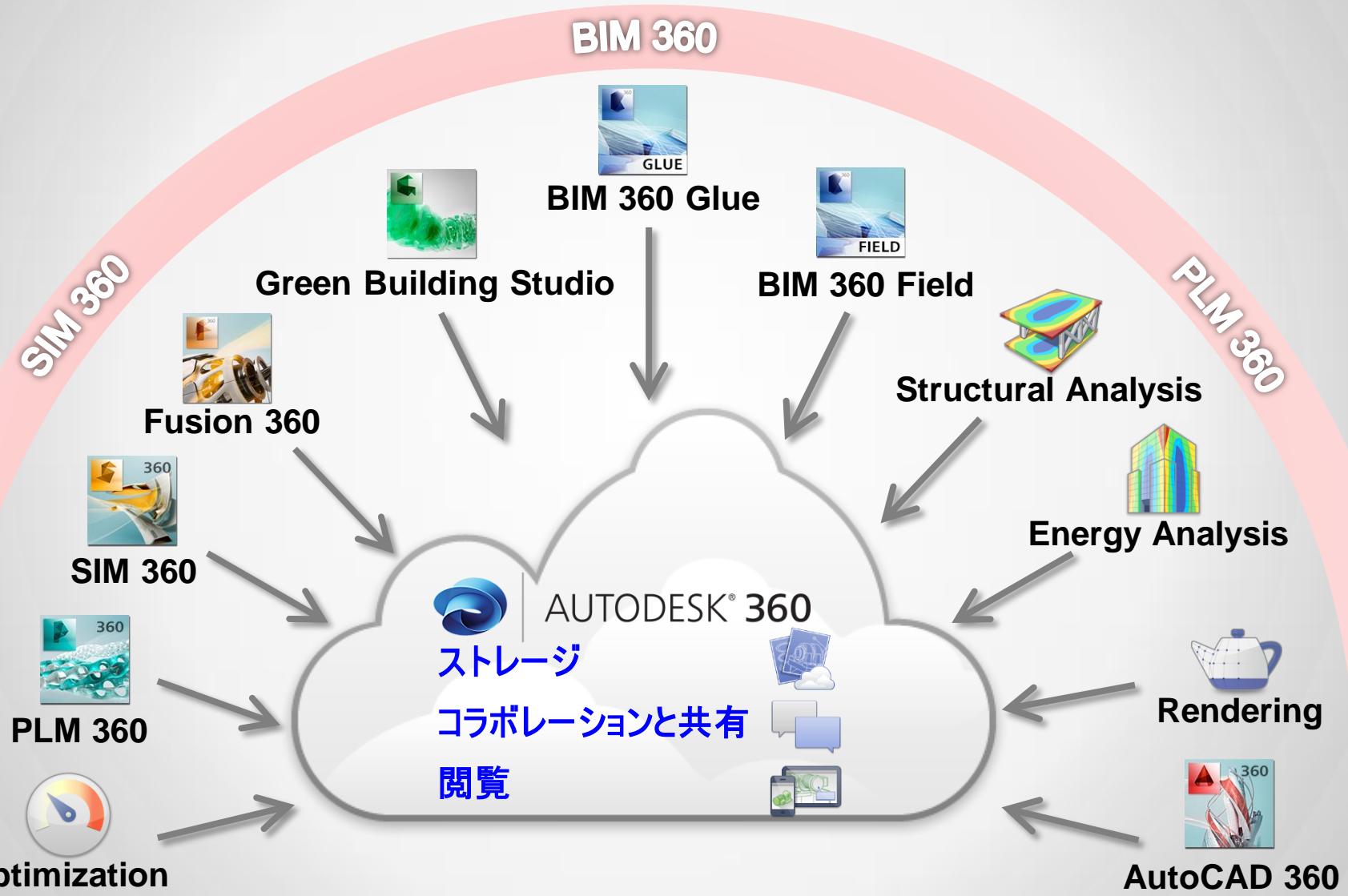
Developer Technical Services



# クラウド&モバイルの活用



# Autodesk 360 クラウド サービス



# クラウドを利用した設計環境へのメリット

## 1. いつでも、どこからでも図面・設計図書にアクセス

- クラウドに保存することでデータ中心の設計環境を実現

## 2. スマートフォン、タブレットなどのモバイル デバイスの活用が可能

- 印刷出力した紙図面の持ち出しが不要、あるいは大幅低減

## 3. 複数の設計者間で図面の共有やオンライン コラボレーションを実現

- 明示的に指定したメンバーでクラウド上の図面を同時に閲覧、協調編集

## 4. クラウドリソースを集中利用してデスクトップPCを演算負荷から解放

- 高負荷なレンダリングや解析演算をクラウドに代替させて手元の作業を継続

# クラウド インフラを提供する代表的な企業

- サービスベンダーに物理的なハードウェアを提供
  - 独自の API を公開してサービスの構築をサポート
  - 多くが世界各地に大規模データセンターを持つ



# クラウドにまつわる用語

データセンター 密結合 シンクライアント IaaS

サービス指向アーキテクチャ 仮想化 データベース サービス

HTTP ホスティング サービス プライベート クラウド

ストレージ サービス Web サービス グリッド コンピューティング

疎結合 SOAP REST オンプレミス システム

ファイル 同期型 サービス

分散 システム インターネット 非同期

アプリケーション サービス

ファイル 共有 サービス

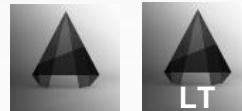
ハイブリッド クラウド

PaaS SaaS

パブリック クラウド

# AutoCAD 360へのアクセス ポイント

- 従来通り 3 つの方法でアクセスできます



デスクトップ製品から



AutoCAD 360 Web

Web ブラウザから



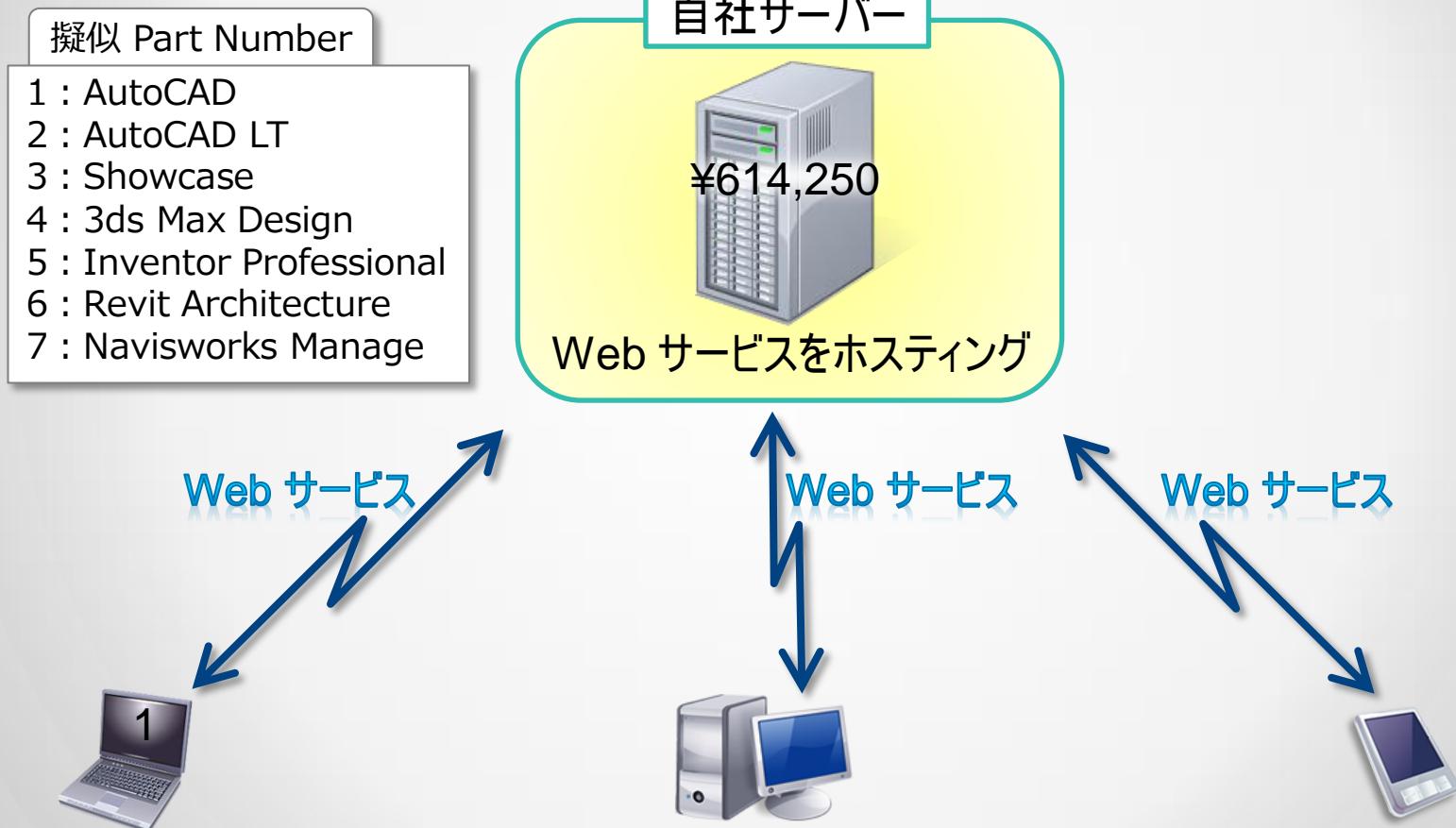
AutoCAD 360 Mobile

Mobile アプリから



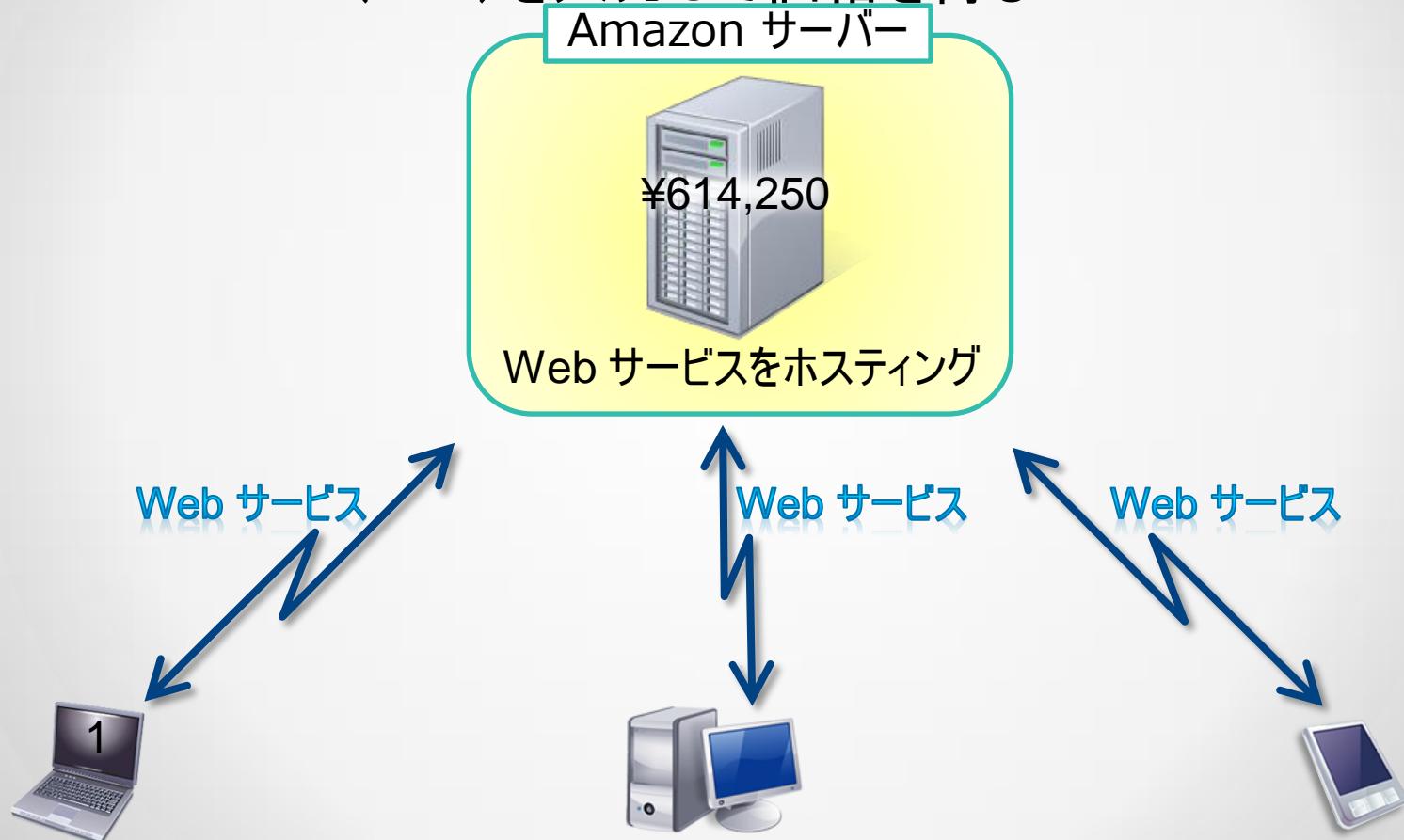
# ① ローカル Web サイトにホストした Web サービス

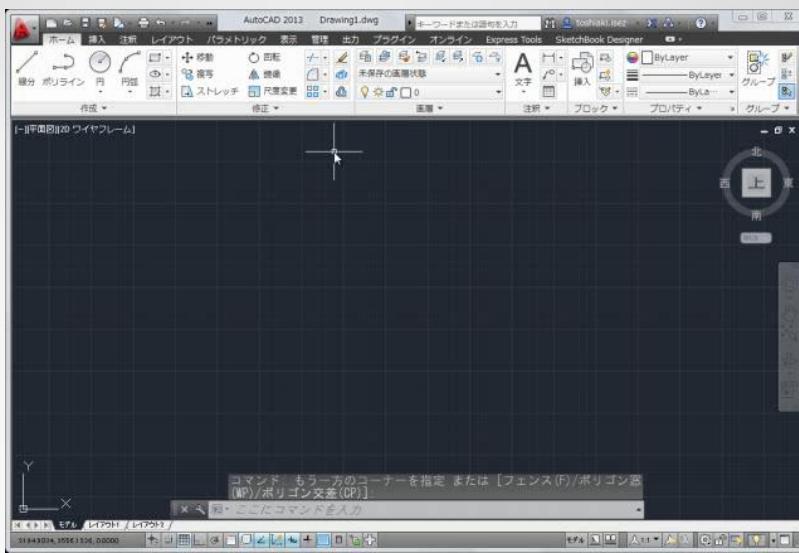
- 想定: Parts Number(P/N)を入力して価格を得る



## ② AWS仮想サーバー上に構築した Web サイトにホストした Web サービス

- Parts Number(P/N)を入力して価格を得る



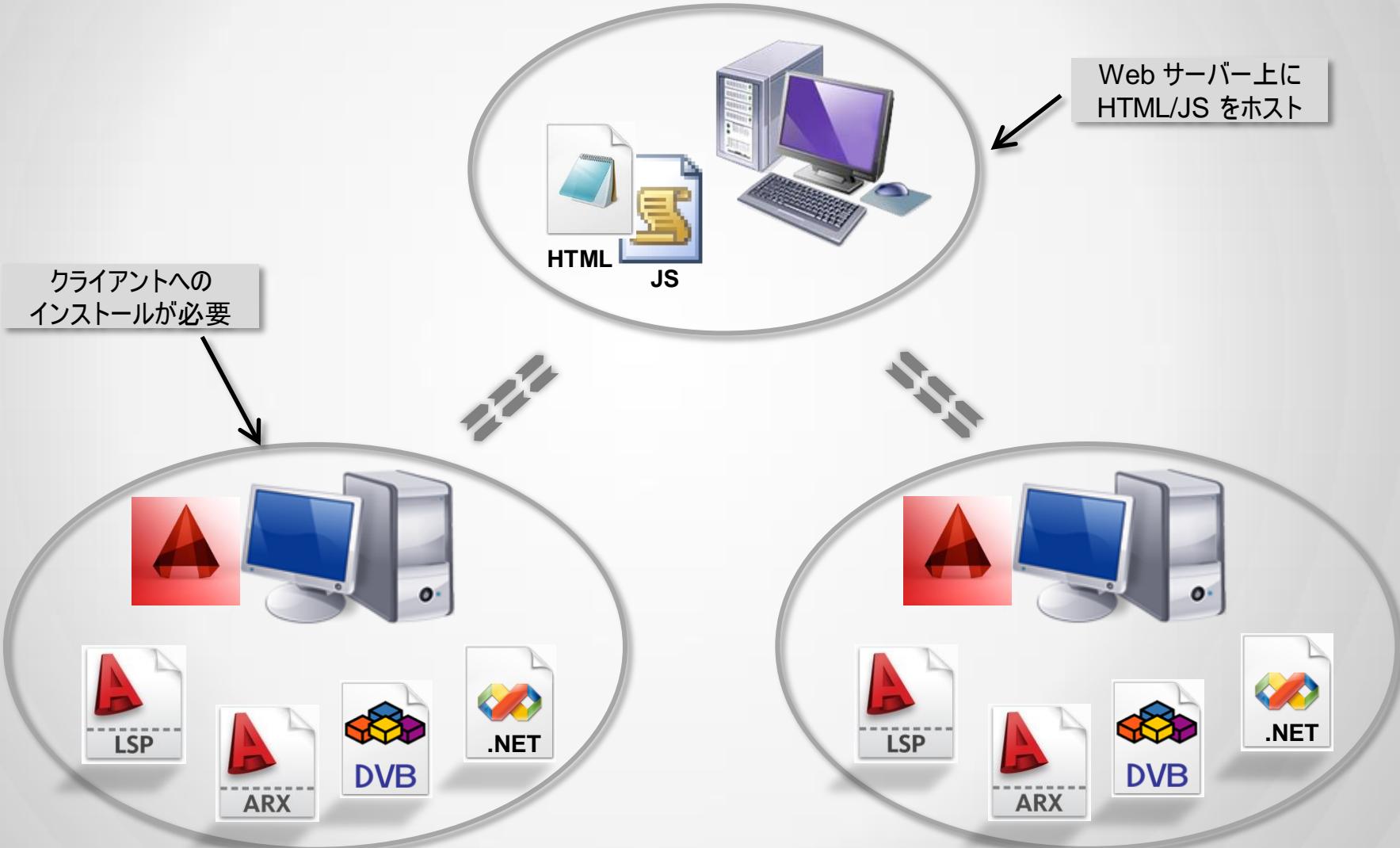


# 参考: 実際の Web サービス例

- Web サービス API の例
  - じゃらん Web サービス:  
<http://www.jalan.net/jw/jwp0000/jww0001.do>
  - カーセンサーnet Web サービス:  
<http://webservice.recruit.co.jp/carsensor/>
  - Google Maps API:  
<https://developers.google.com/maps/>
  - GeOAP:  
<http://www.geoap.jp/>
- Web サービス API を持つオートデスク製品
  - AutoCAD BIM 360 Gule
  - ...

# クラウドとモバイル – AutoCAD JavaScript API

## サンプル



# クラウドとモバイル – AutoCAD 360

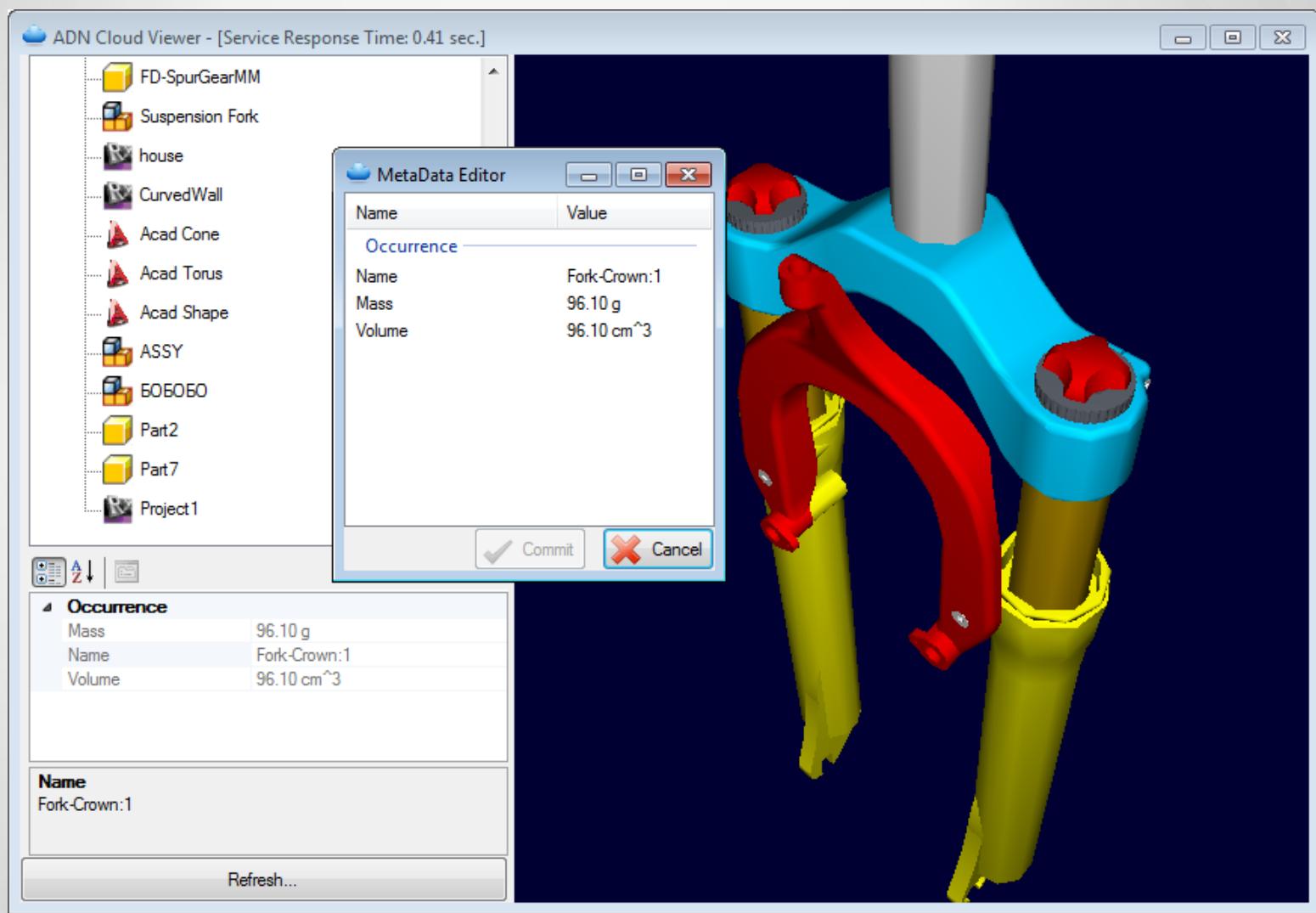
サンプル



This screenshot shows another web browser window for the AutoCAD 360 support center. The URL is <https://autocad360.desk.com/customer/ja/portal/articles/1089127-autocad-360-を-web-サイトに埋め込む>. The page title is 'AutoCAD 360 を Web サイトに埋め込む'. It provides instructions for embedding AutoCAD 360 applications into a website using an IFRAME HTML tag. It includes a sample code snippet and a note about embedding specific drawings. There are also sections for reporting bugs and sending email.

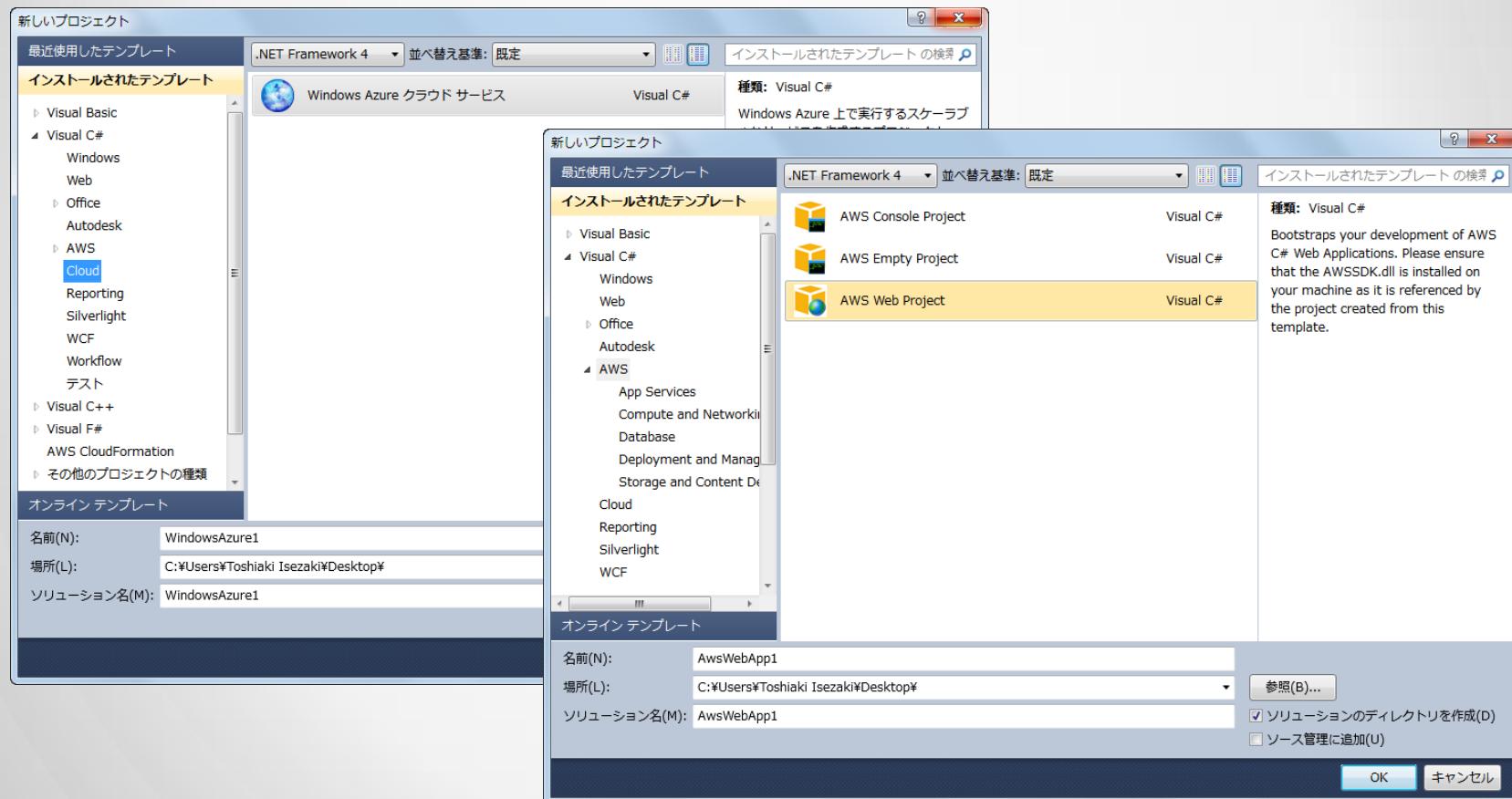
# クラウドとモバイル－独自開発

サンプル

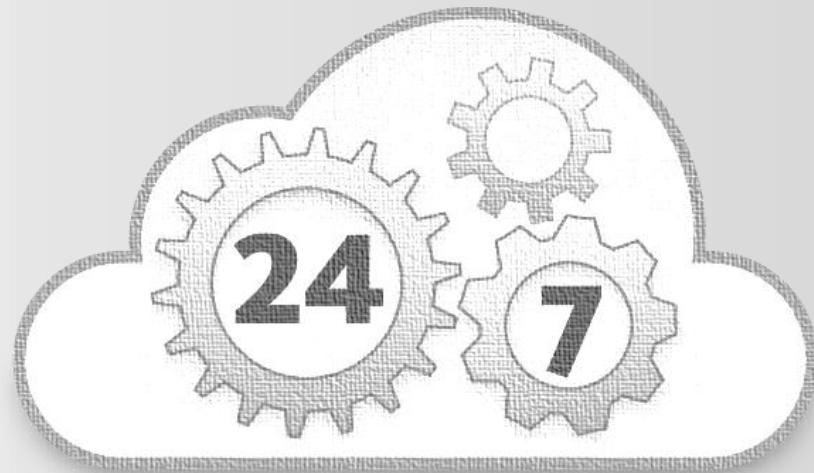


# どのように開発すればよいのか？

- クラウド ベンダーによる支援ツールを利用可能

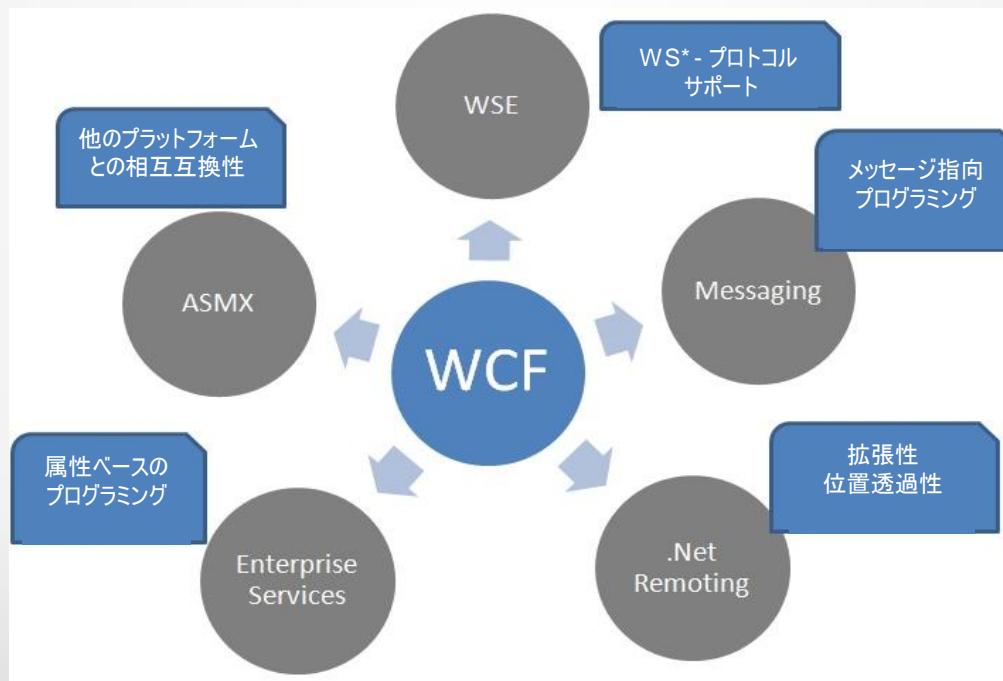


# WCF Web サービスの実装



# WCF とは ?

- **WCF (Windows Communication Foundation) とは**
  - Microsoft の次世代プログラミング プラットフォームでバインディング、設定、ビルド、ネットワーク配布されるサービスの展開のためのランタイム システムです(出典 MSDN)



# WCF の基礎 - ABC

- “ABC” はどう WCF サービスが構成されるか理解する鍵:

✓ “A” は **Address** を表現:

サービスはどこにあるのか ?

✓ “B” は **Binding** を表現:

クライアントはどうサービスと会話するのか ?

✓ “C” は **Contract** を表現:

サービスはクライアントに何ができるのか ?

# データ コントラクト

```
[DataContract]
public class AdnMaterial
{
    [DataMember]
    public string Name
    {
        get;
        set;
    }

    [DataMember]
    public double Price
    {
        get;
        set;
    }

    [DataMember]
    public string Manufacturer
    {
        get;
        set;
    }
}
```

# サービスコントラクト

```
[ServiceContract]
public interface IMaterialSrv
{
    [OperationContract]
    AdnMaterial[] GetMaterials();

    [OperationContract]
    AdnMaterial GetMaterial(
        string materialName);

    [OperationContract]
    bool PostMaterial(
        AdnMaterial material);
}
```

# サービスの実装

```
public class MaterialSrv : IMaterialSrv
{
    public MaterialSrv()
    {
        //Constructor...
    }

    public AdnMaterial[] GetMaterials()
    {
        //Implementation...
    }

    //Implementation other methods...
}
```

# サービスの動作

```
[ServiceBehavior(  
    InstanceContextMode=InstanceContextMode.PerSession,  
    TransactionTimeout="00:00:30")]  
public class MaterialSrv : IMaterialSrv  
{  
    [OperationBehavior(  
        TransactionAutoComplete=true,  
        TransactionScopeRequired=false)]  
    public AdnMaterial[] GetMaterials()  
    {  
    }  
}
```

# サービスのバインディング

- バインディングはアドレスにサービス コントラクトの実装をバインド
- 次の項目を含む:
  - 適切な転送プロトコルを選択
  - どうメッセージがエンコードされるかを選択
  - どうセキュリティが動作するかを定義

# サービスのホスティング

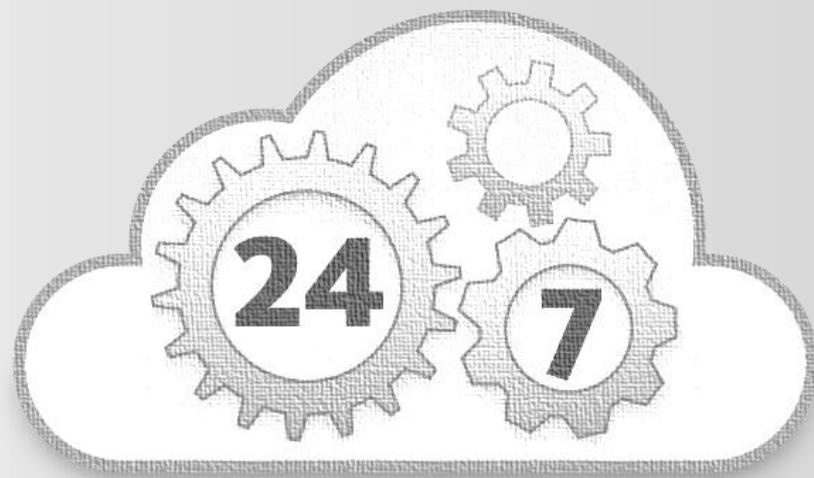
- WCF サービスをホスティングするために利用可能なオプション:
  - マネージ .NET アプリケーションのセルフ ホスティング
    - Winform、WPF、コンソール アプリケーション、 ...
  - Windows サービスでのホスティング
  - IIS でのホスティング
    - ASP.Net
    - ダイレクト ホスティング

# REST サポートの追加

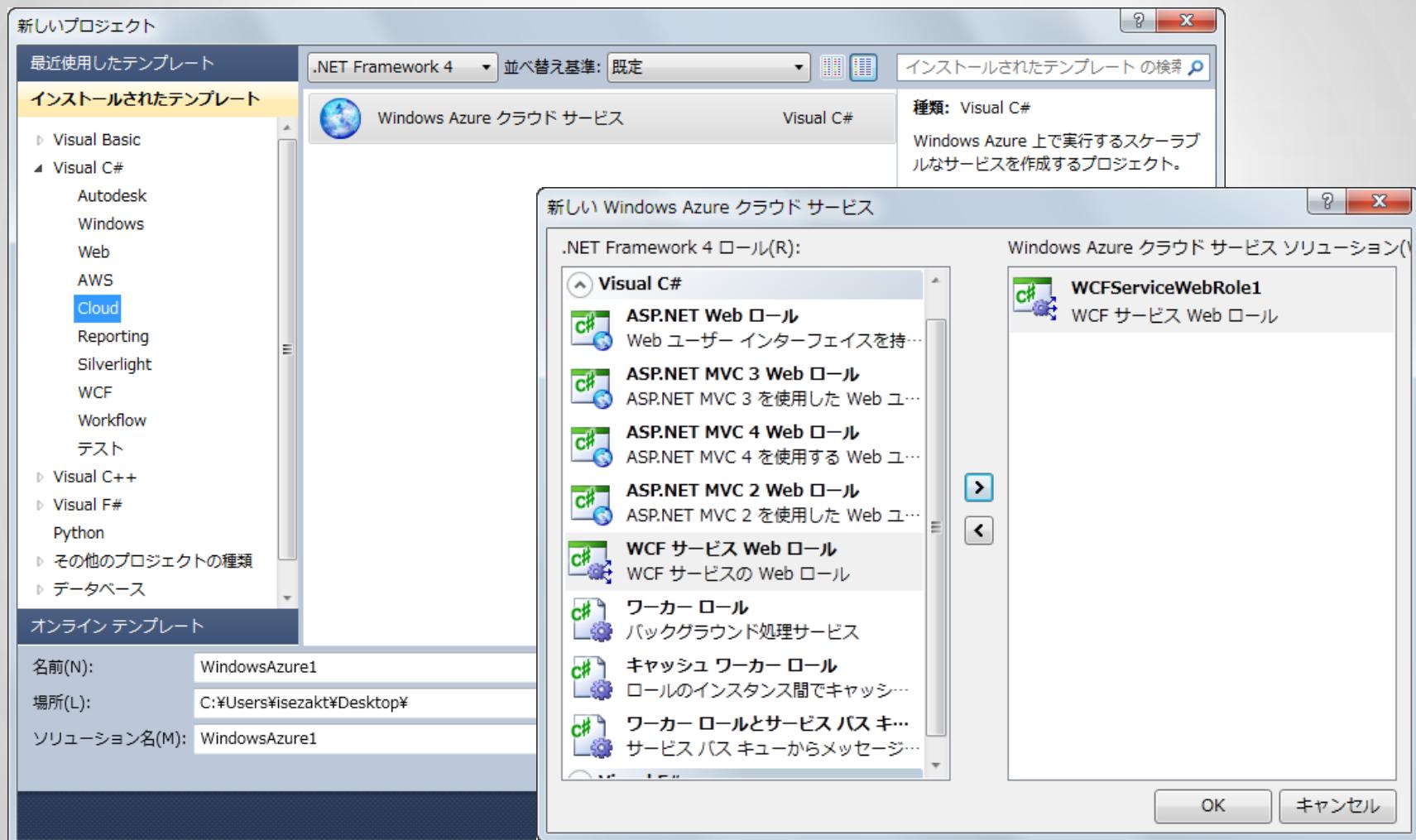
```
[ServiceContract]
public interface IMaterialSrv
{
    [OperationContract]
    [WebInvoke(
        Method = "GET",
        UriTemplate = "/Material/{materialName}",
        ResponseFormat = WebMessageFormat.Json)]
    AdnMaterial GetMaterial(string materialName);

    [OperationContract]
    [WebInvoke(
        Method = "POST",
        UriTemplate = "/Material",
        RequestFormat = WebMessageFormat.Json,
        ResponseFormat = WebMessageFormat.Json)]
    bool PostMaterial(AdnMaterial material);
}
```

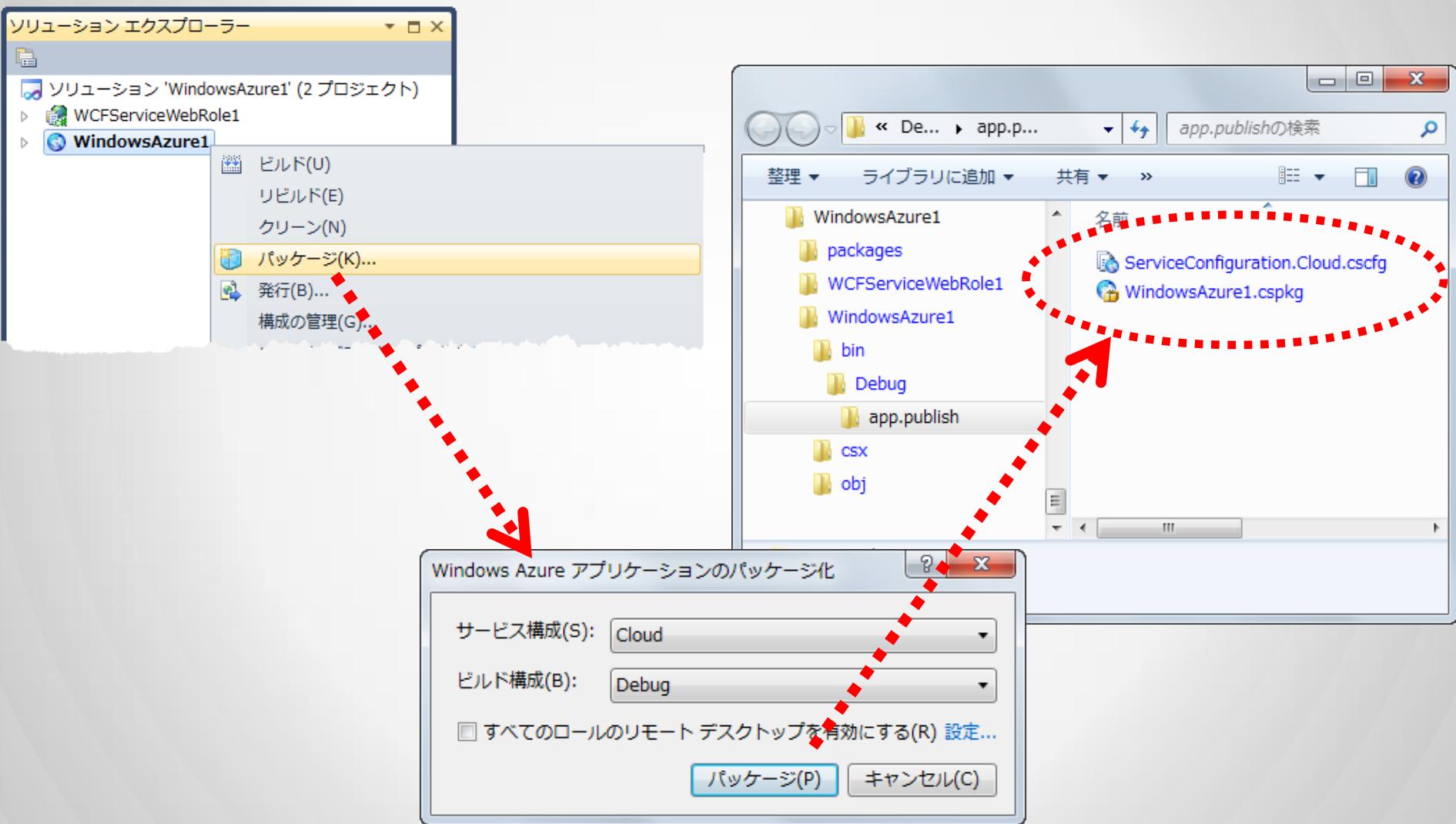
# クラウドへのデプロイメント Windows Azure の例...



# Step 1 - Azure プロジェクト



# Step 2 – パッケージ作成



# Step 3 - Azure のデプロイ

The screenshot shows the Windows Azure portal interface. On the left, a sidebar lists various service categories: 'すべてのアイテム', 'クラウド サービス' (0), 'SQL データベース' (0), 'ストレージ' (0), 'サービスバス' (0), 'アドオン' (0), '設定', and a prominent '新規' button highlighted with a red dashed circle. The main area is titled '新しいクラウド サービス - カスタム作成' and displays the 'Cloud Service Creation Wizard'. The first step, 'Cloud Service Deployment', is shown with the sub-step 'Cloud Service Creation'. The 'Name' field contains 'adnmaterial'. Other tabs include 'Service Status', 'Deployment', 'Subscription', 'Location', and 'URL'. Under 'Deployment', the package 'AdnMaterialSrv.cspkg' is selected from 'Local' storage. The 'Configuration' section shows 'ServiceConfiguration.Cloud.cscfg' selected from 'Local' storage. The 'Environment' section has 'Deployment' selected. At the bottom, there are three checkboxes: '1つ以上のロールに単一のインスタンスが含まれている場合でも展開します。' (checkbox checked), '展開の開始' (checkbox checked), and '今すぐ証明書を追加' (checkbox unchecked). Navigation buttons 'Back' and 'Next' are at the bottom right.

# Step 4 – テスト

Fiddler - HTTP Debugging Proxy

File Edit Rules Tools View Help GET /book

Replay Resume Stream Decode Keep: All sessions iexplore:5572 Find Save Browse Clear Cache TextWizard Tearoff

Web Sessions

#	Result	Protocol	Host	URL	Body	Cac
5	200	HTTP	admmaterial.clouda...	/MaterialSrv.svc/r...	600	

Statistics Inspectors AutoResponder Composer Filters Log Timeline

Headers TextView WebForms HexView Auth Cookies Raw JSON XML

Request Headers

GET /MaterialSrv.svc/rest/getmaterials HTTP/1.1

[Raw] [Header Definitions]

Client

```
Accept: text/html, application/xhtml+xml, */*
Accept-Encoding: gzip, deflate
Accept-Language: en-US
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)
```

Transport

```
Connection: Keep-Alive
```

Get SyntaxView Transformer Headers TextView ImageView HexView WebView Auth

Caching Cookies Raw JSON XML

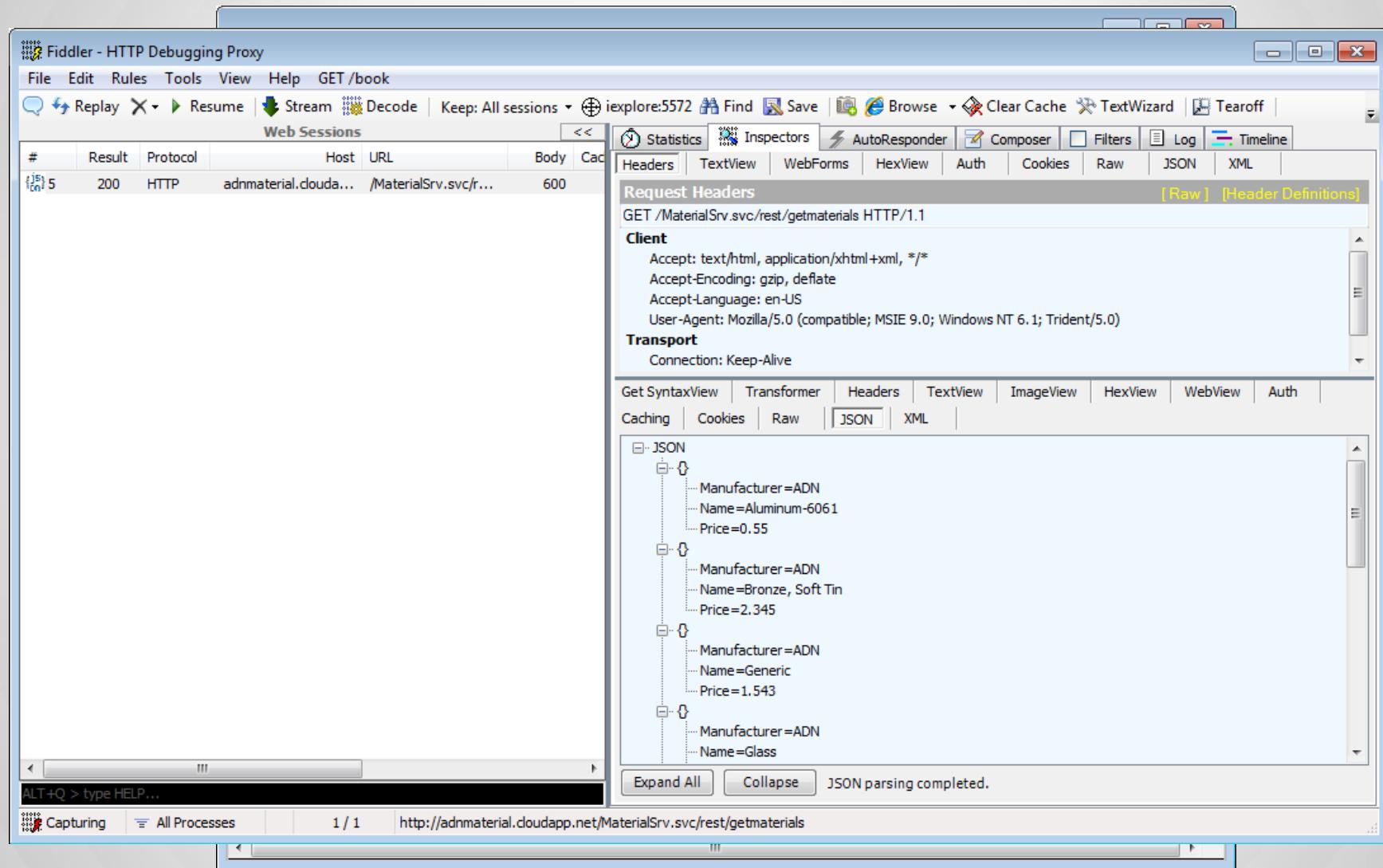
JSON

- Manufacturer=ADN  
  - Name=Aluminum-6061  
  - Price=0.55
- Manufacturer=ADN  
  - Name=Bronze, Soft Tin  
  - Price=2.345
- Manufacturer=ADN  
  - Name=Generic  
  - Price=1.543
- Manufacturer=ADN  
  - Name=Glass

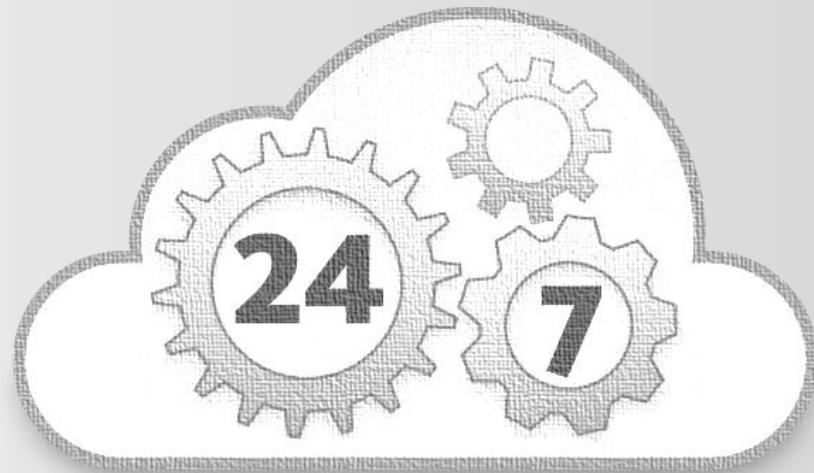
Expand All Collapse JSON parsing completed.

ALT+Q > type HELP...

Capturing All Processes 1 / 1 http://admmaterial.cloudapp.net/MaterialSrv.svc/rest/getmaterials

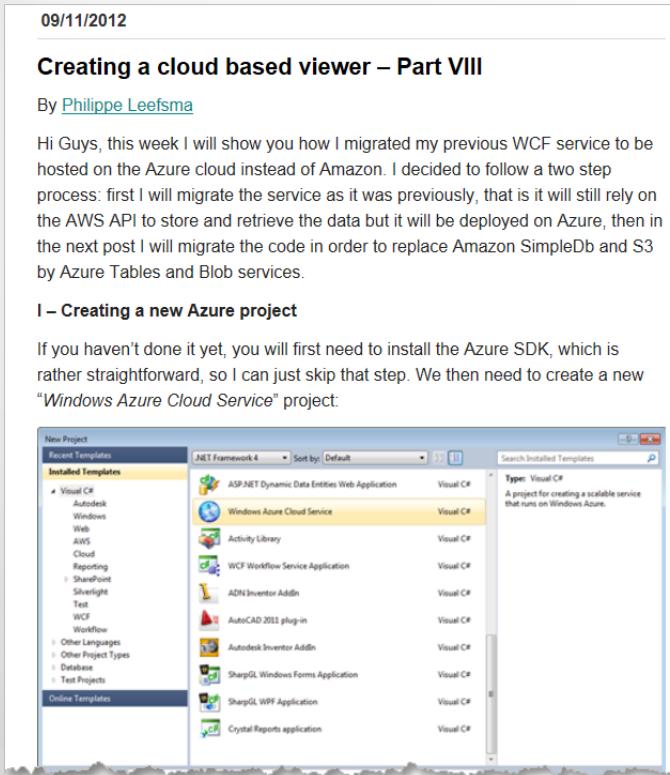


# 学習リソース



# アドオン開発者のためのクラウド&モバイルの開発情報

- ブログ記事で公開:
  - [http://adndevblog.typepad.com/cloud\\_and\\_mobile/](http://adndevblog.typepad.com/cloud_and_mobile/)



10/22/2012

## Real Cross-Platform Interactive 2D Graphics Using SVG

By [Jeremy Tammik](#).

Hi everybody.

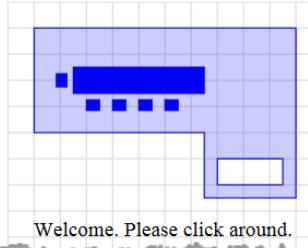
This is my first post to the Cloud and Mobile DevBlog.

All I will do here is point out that I spent my education day last Friday exploring how to display and interact with 2D graphics in a browser using SVG and JavaScript by implementing a very simple [room polygon and furniture picker in SVG](#).

The main goal was to enable picking and identifying objects and ensuring that the same code and interaction is supported on all mobile devices, in particular both Android and Safari on iOS.

The latter added an additional little challenge at the end.

Here is the final result:



Welcome. Please click around.

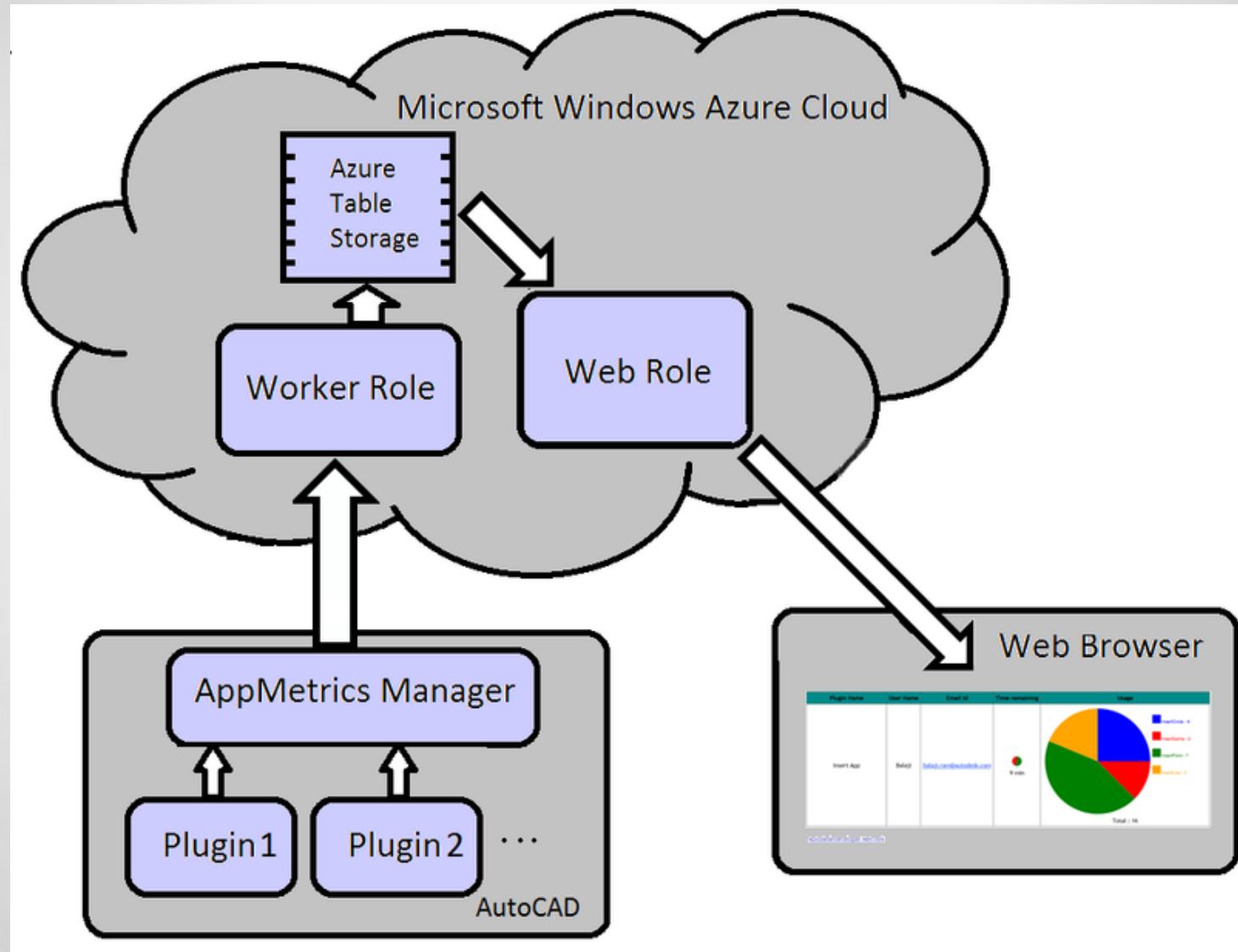
# AutoCAD JavaScript API

- ブログ記事で公開:
  - [http://adndevblog.typepad.com/technology\\_perspective/2013/05/autocad-javascript-api-part1.html](http://adndevblog.typepad.com/technology_perspective/2013/05/autocad-javascript-api-part1.html)
  - [http://adndevblog.typepad.com/technology\\_perspective/2013/05/autocad-javascript-api-part2.html](http://adndevblog.typepad.com/technology_perspective/2013/05/autocad-javascript-api-part2.html)
  - [http://adndevblog.typepad.com/technology\\_perspective/2013/06/autocad-javascript-api-part3.html](http://adndevblog.typepad.com/technology_perspective/2013/06/autocad-javascript-api-part3.html)
  - [http://adndevblog.typepad.com/technology\\_perspective/2013/07/autocad-2014-javascript-api\\_development\\_prep.html](http://adndevblog.typepad.com/technology_perspective/2013/07/autocad-2014-javascript-api_development_prep.html)

# 公開したアドオンアプリのライセンス管理のアイデア

- ブログ記事でアイデアを公開:
  - <http://adndevblog.typepad.com/autocad/2012/08/licensing-applications-infralution.html>
  - <http://adndevblog.typepad.com/autocad/2012/09/licensing-applications-crypto.html>
  - <http://adndevblog.typepad.com/autocad/2012/10/trial-licensing-and-usage-metrics-gathering-for-autocad-plugin-using-azure-cloud.html>

# クラウドを利用した期限付きライセンスの実現





Autodesk is a registered trademark of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and/or other countries. All other brand names, product names, or trademarks belong to their respective holders. Autodesk reserves the right to alter product offerings and specifications at any time without notice, and is not responsible for typographical or graphical errors that may appear in this document.