



Revit API & BIM セミナー 2018

# Revit アドイン基礎

伊勢崎 俊明

Autodesk Developer Network / Forge Partner Development



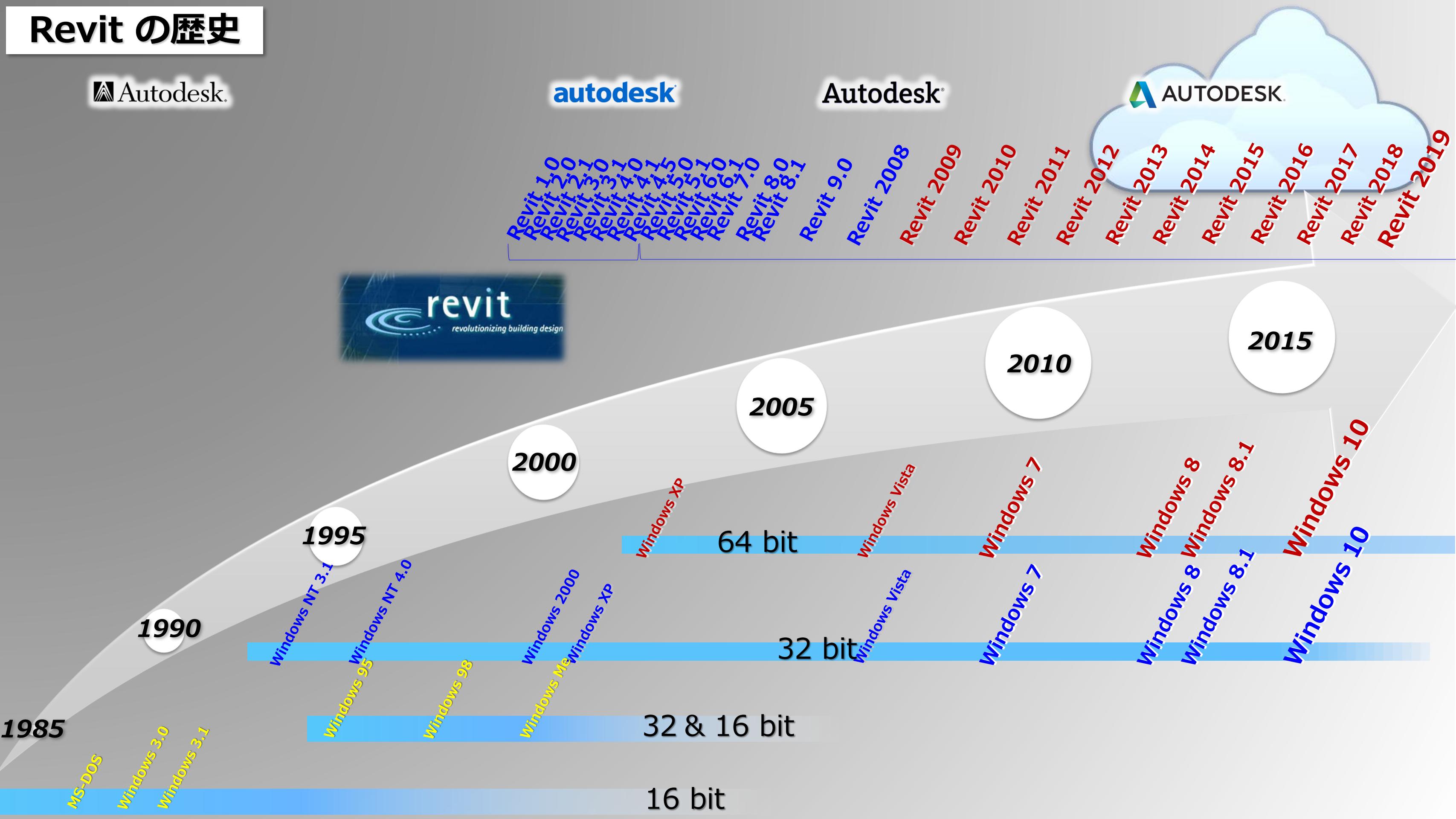
# アジェンダ – ご紹介する内容

- Revit のカスタマイズ手法
- Revit アドイン開発環境
- 補足 & ご案内

# Revit のカスタマイズ手法



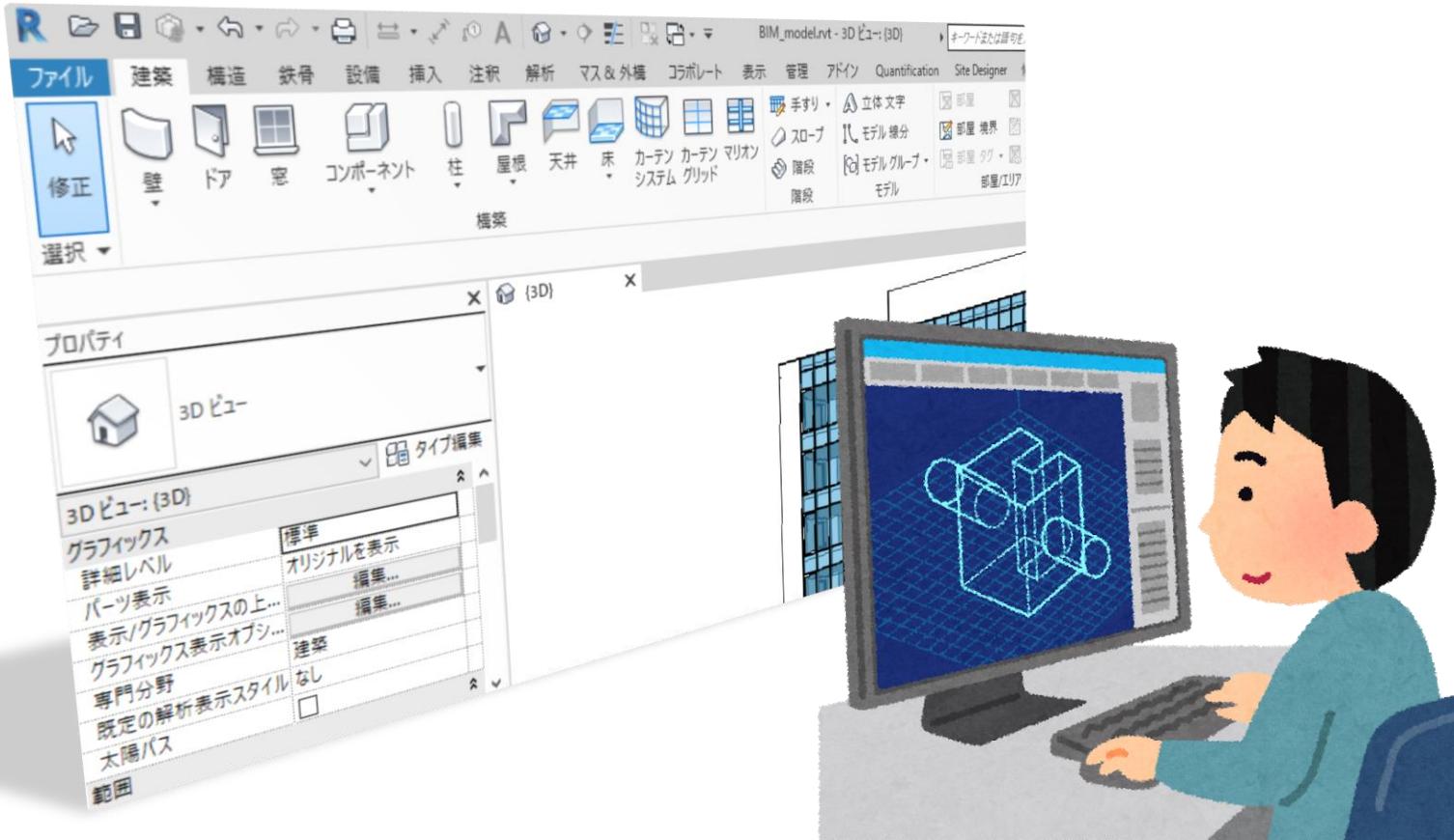
# Revit の歴史



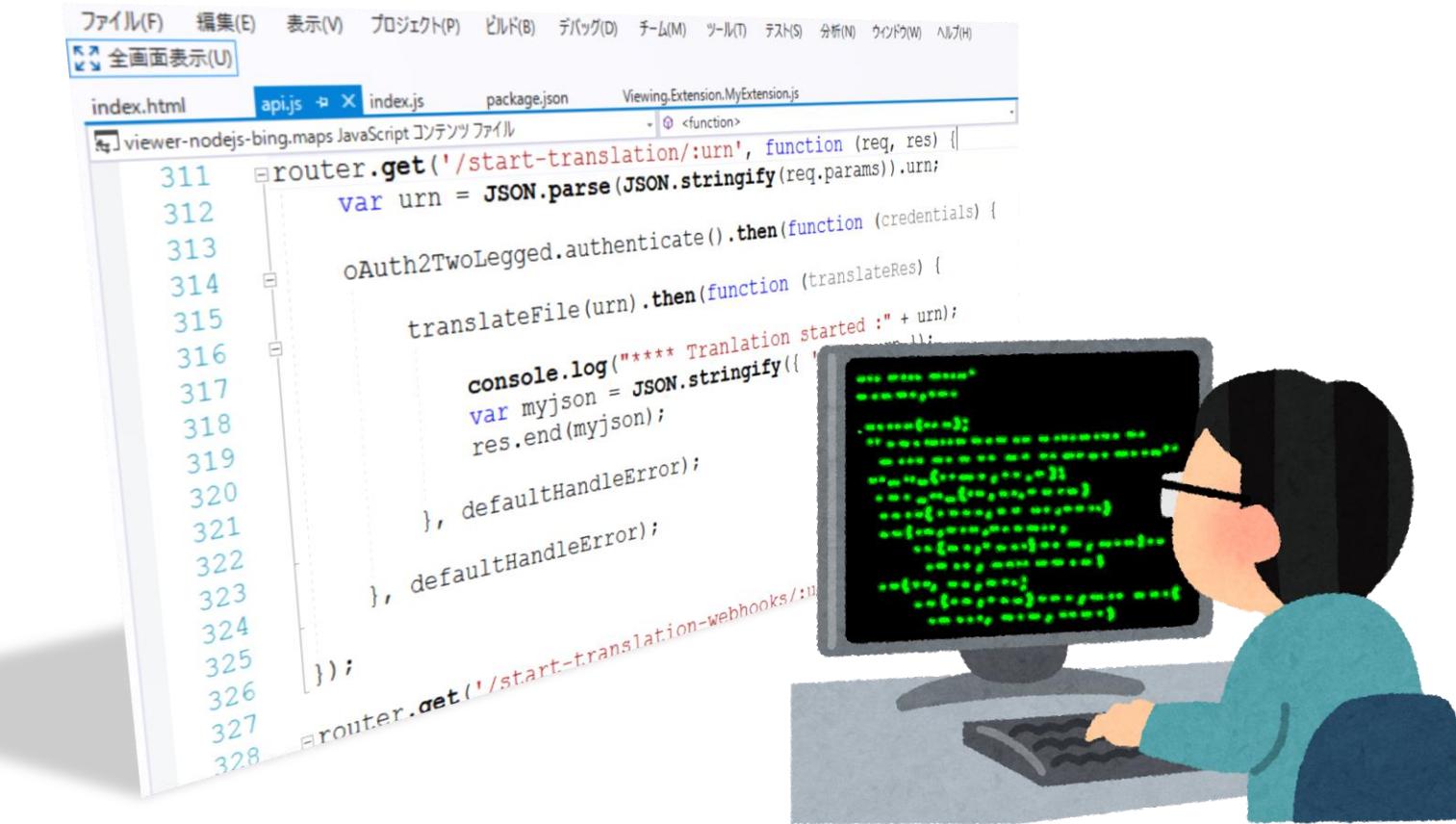
# ソフトウェア クラウド サービス

2つのアクセス方法

## Graphical User Interface



## Application Programming Interface



# Revit プログラミングの種類

- **Dynamo for Revit**

- ビジュアルプログラミングでスクリプトを作成
- 製品組み込み開発環境



- **SharpDevelop**

- プログラム言語を使ってマクロを作成
- 製品組み込み開発環境

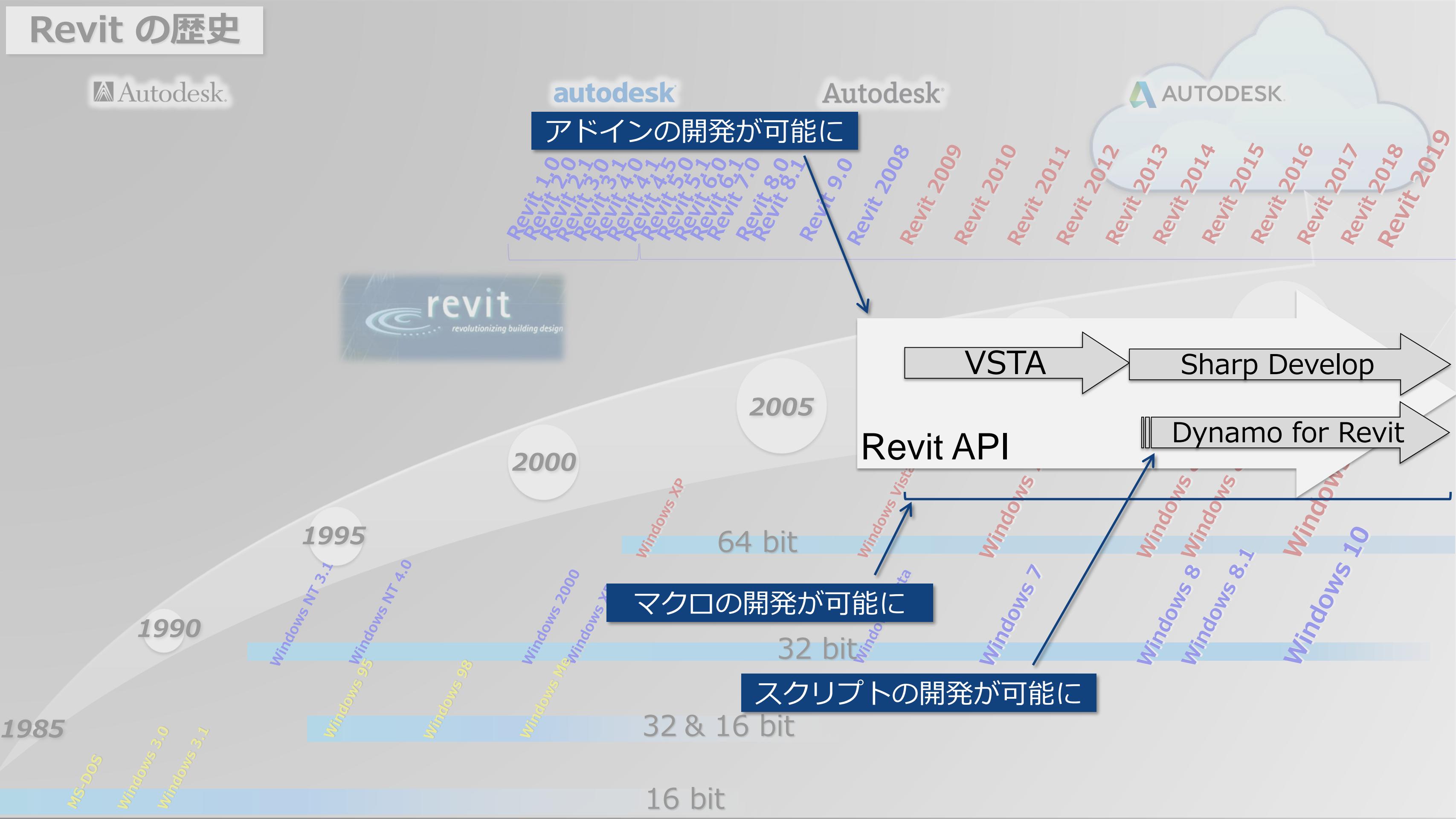


- **Visual Studio (別売り)**

- プログラム言語を使ってアドインを作成
- 別名、アドオン、プラグイン

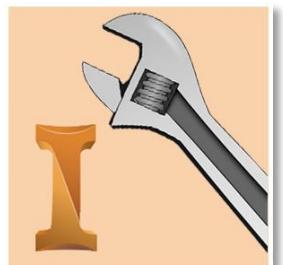
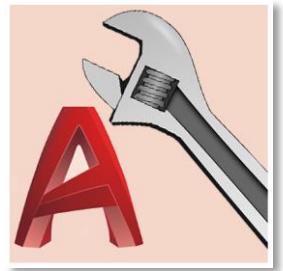


# Revit の歴史



# Revit 機能を API として公開しているテクノロジ

- AutoCAD
  - AutoLISP : AutoCAD 機能として独自実装
  - ObjectARX : ネイティブ C++
  - ActiveXオートメーション : Component Object Model
  - AutoCAD .NET API : .NET Framework
- Revit
  - Revit API : .NET Framework
- Inventor
  - Inventor API : Component Object Model

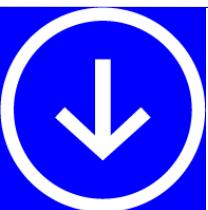


# .NET Framework が持つ 2 つの側面



## 機能を公開する役割

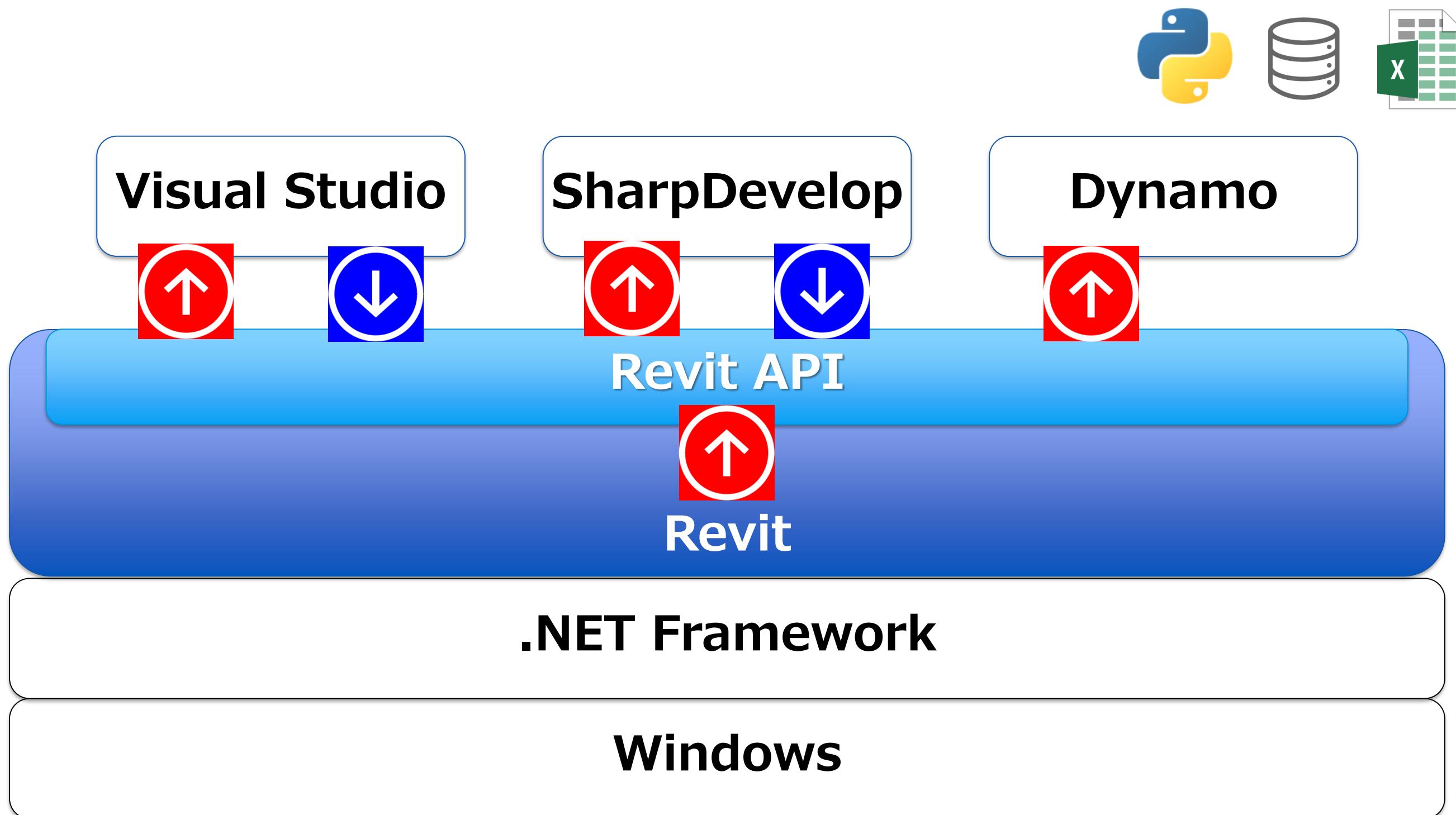
- アセンブリ ファイル (.dll) を使って公開
- Revit の機能を API として公開
  - **RevitAPI.dll**、**RevitAPIUI.dll**、...
  - Revit インストール フォルダにインストールされている



## 公開された機能を利用する役割

- 開発環境 (IDE) に読み込んで利用 - プログラムに活用
- IDE でビルドしてアセンブリ ファイル (.dll) を作成  
→ アドイン ファイル

# Revit プログラミング環境



# Revit プログラミング環境

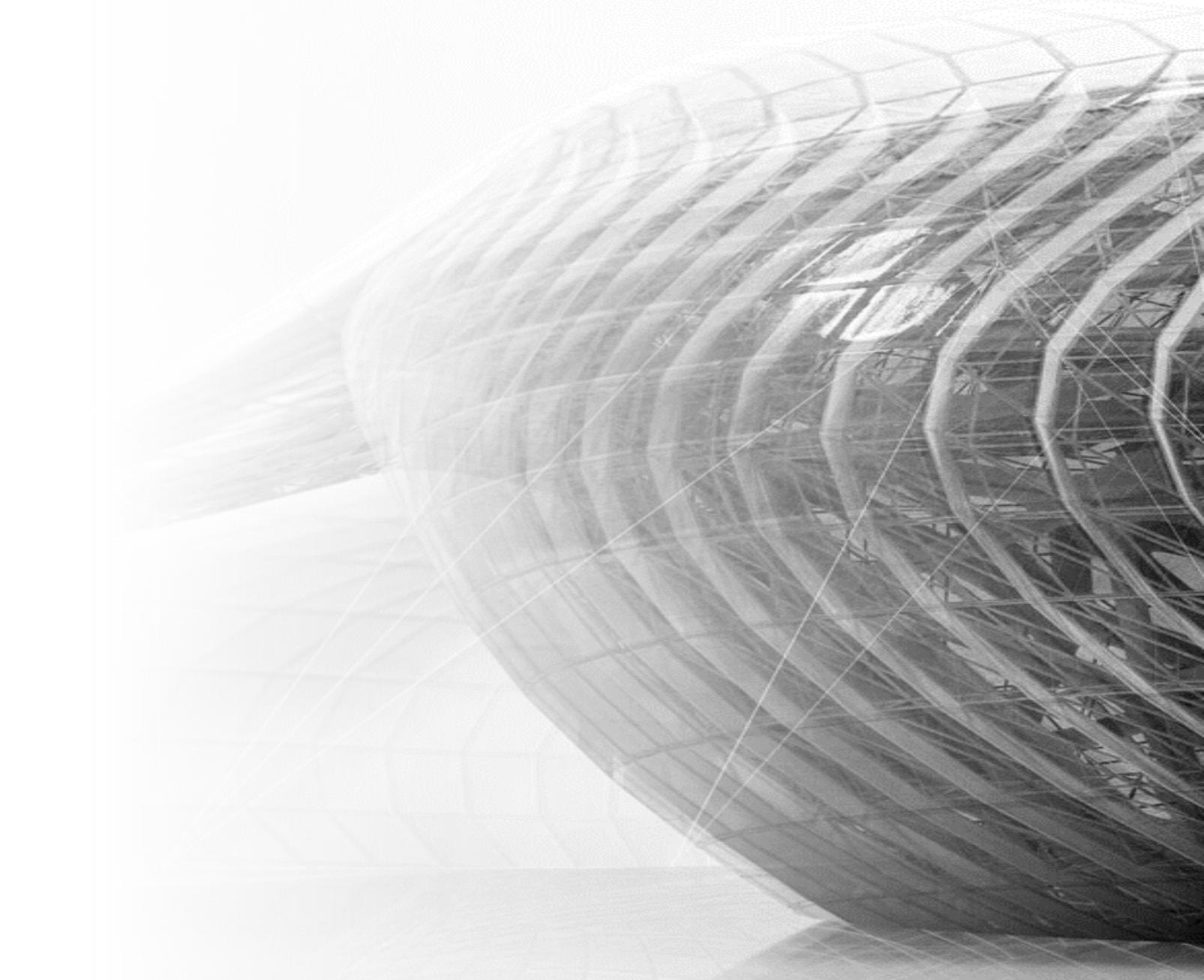
	アドイン	本セミナー	マクロ	Dynamo スクリプト
プログラミング	開発言語 C#, VB.NET, C++/CLI, Ruby <sup>*1</sup> , Python <sup>*2</sup>		開発言語 C#, VB.NET, Ruby, Python	ビジュアル Dynamo グラフ
コンパイル	○		○	×
要素作成	○		○	○
要素編集	○		○	○
要素選択	○		○	○
プロパティ取得	○		○	○
プロパティ編集	○		○	○
パラメータファイル読み込み <sup>*3</sup>	×		×	○
パラメータファイル書き込み <sup>*3</sup>	×		×	○
Revit UI ボタン、リボン	○		×	×
拡張性	.NET, COM Interop		.NET, COM Interop	Design Script, Python

\*1 IronRuby

\*2 <https://docs.microsoft.com/ja-jp/visualstudio/python/installing-python-support-in-visual-studio>

\*3 Revit 要素に連動するパラメータ/プロパティの Excel や CSV ファイルの入出力

# Revit アドイン開発



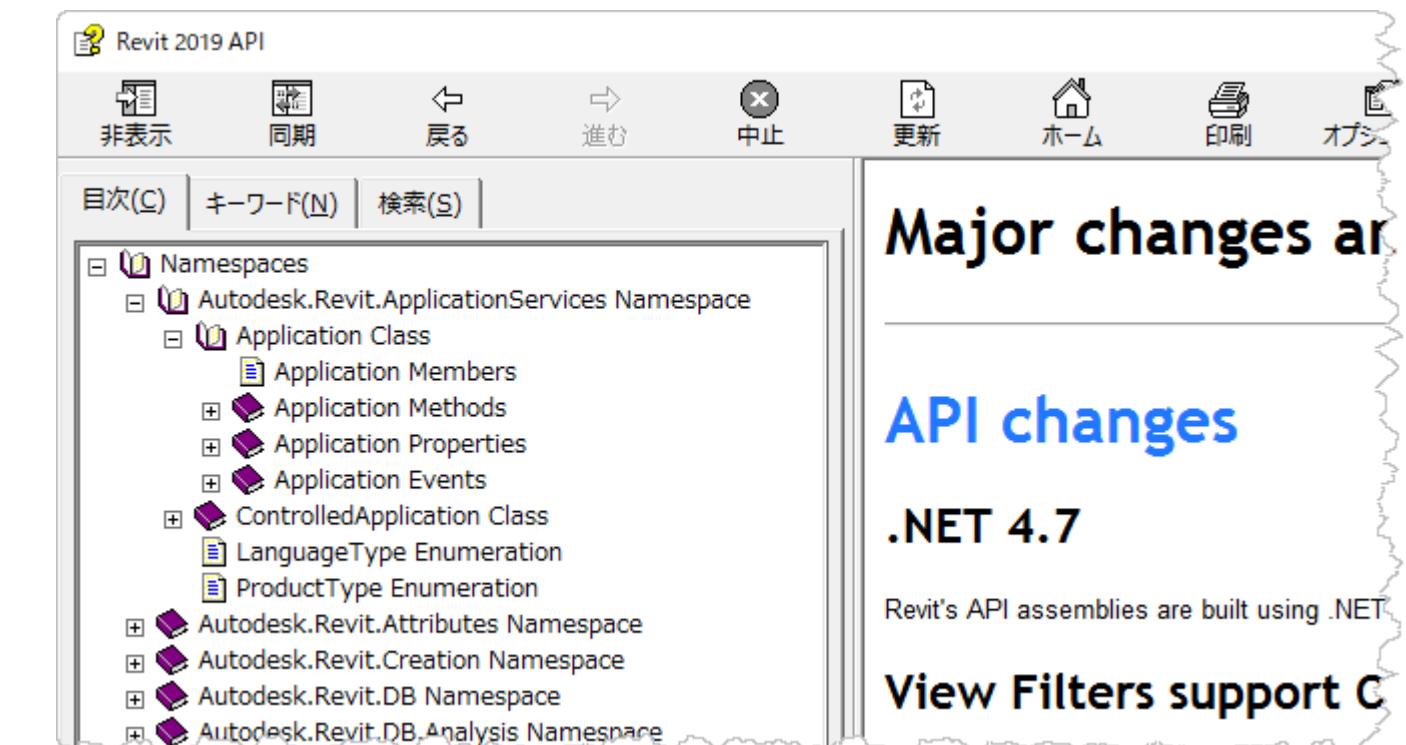
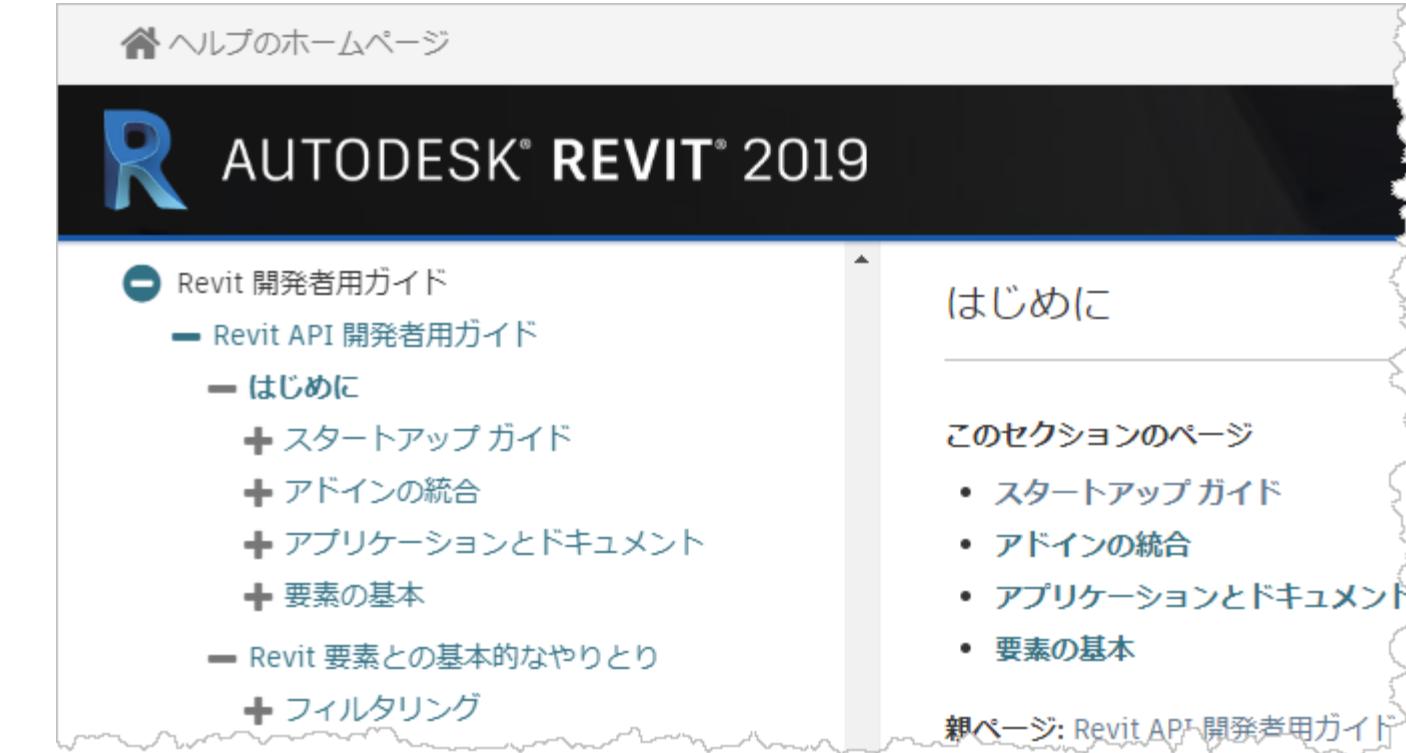
# Revit と .NET Framework バージョン

- アドイン アセンブリのターゲットバージョン
  - Revit 2019 .NET Framework 4.7
  - Revit 2018 .NET Framework 4.6
  - Revit 2017 .NET Framework 4.6※4.5.2 許容
  - Revit 2016 .NET Framework 4.5
  - Revit 2015 .NET Framework 4.5
  - Revit 2014 .NET Framework 4.0 Update1
  - Revit 2013 .NET Framework 4.0 Update1
- 利用可能な Visual Studio
  - 当該バージョンに一致するアセンブリ出力が可能なものの



# Revit 2019 アドイン開発で参照すべきドキュメント

- Revit 開発者用ガイド（日本語）
  - 製品オンラインヘルプ内
- Revit API リファレンス（英語）
  - Revit SDK 内



# オンラインヘルプ<sup>®</sup>推奨されるアドイン開発環境



- **Visual Studio 2015 Professional**

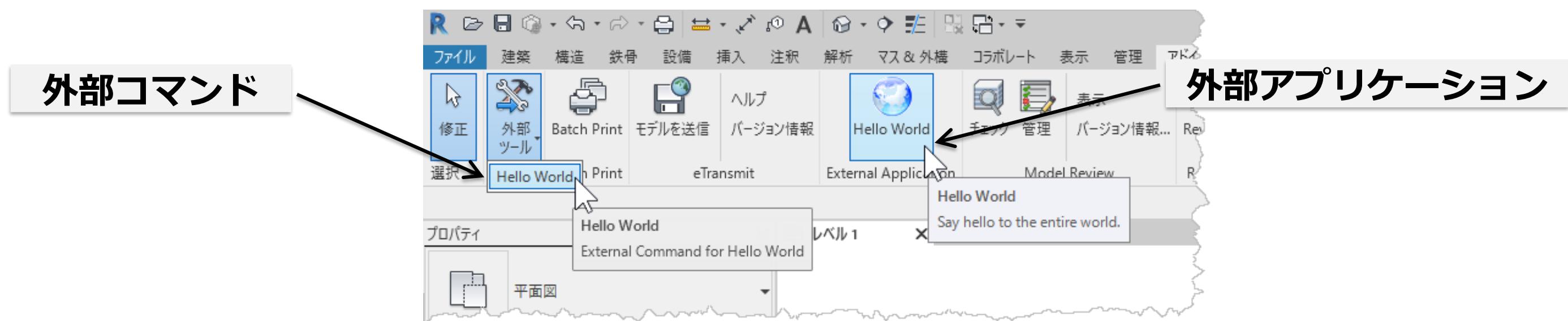
- Express Edition（無償）はデバッグ機能がないため非推奨
- プロジェクトのターゲットフレームワークは 4.7 に設定
- **Visual Studio 2017** もターゲットフレームワーク 4.7 設定可
- 開発言語は C#、VB.NET、C++/CLI（マネージC++）が可能
- ほとんどの方が C# を選択（実績が豊富なので C# がお勧め）

# アドイン作成手順概要

1. Visual Studio でクラス ライブラリプロジェクトを新規作成
2. Revit API 公開アセンブリを参照設定
3. ターゲット フレームワークを 4.7 に設定
4. コーディング
5. ビルド (コンパイル)
6. Revit にビルドしたアセンブリ (.dll ファイル) をロード
7. ロード用にマニフェスト (.addin ファイル) を作成、配置

# アドインを使った Revit の拡張

- 外部コマンド
  - IExternalCommand インタフェースを利用する実装
  - コマンドは [アドイン] リボンの [外部ツール] に追加される
- 外部アプリケーション
  - IExternalApplication インタフェースを利用する実装
  - アプリケーションは [アドイン] リボンにパネル/タブ作成可能
  - 外部アプリケーションは外部コマンドを呼び出し可能



# 外部コマンドの実装 (C# 例)

```
[Transaction(TransactionMode.Manual)]
public class Command : IExternalCommand
{
    public Result Execute(
        ExternalCommandData commandData,
        ref string message,
        ElementSet elements)
    {
        TaskDialog.Show("External Command", "Hello World!");
        return Result.Succeeded;
    }
}
```

Revit データ書き込みを宣言するトランザクション モード属性:

- TransactionMode.Manual
- TransactionMode.ReadOnly

← IExternalCommand からのクラス派生

コマンド処理の実態となる Execute メソッド  
(Revit オブジェクトモデルへのアクセス,  
コマンド実行失敗時のユーザ メッセージ,  
コマンド実行失敗時にハイライトする要素)

← Revit へ返す (知らせる) 戻り値:

- Result.Succeeded
- Result.Failed
- Result.Cancelled

↑ タスク ダイアログの表示

# 外部アプリケーションの実装 (C# 例)

```
class App : IExternalApplication
{
    public Result OnStartup(UIControlledApplication application)
    {
        return Result.Succeeded;
    }

    public Result OnShutdown(UIControlledApplication application)
    {
        return Result.Succeeded;
    }
}
```

IExternalApplication からのクラス派生

Revit の起動時に OnStartup が呼び出される

Revit の終了時に OnShutdown が呼び出される

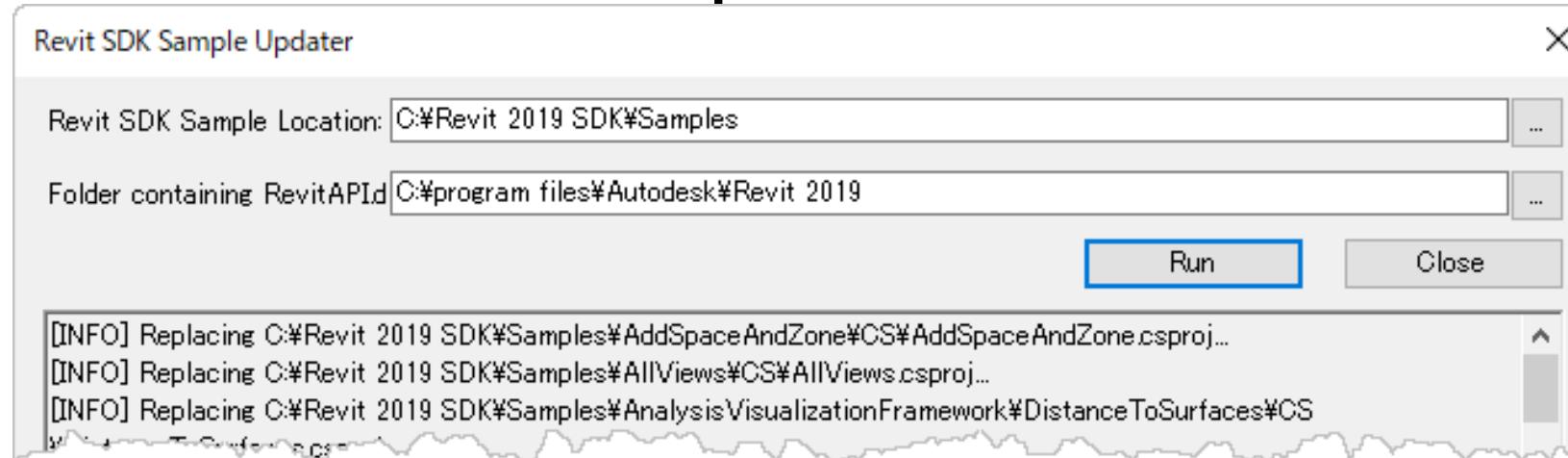
※ ユーザ インタフェースを利用しないタイプの外部アプリケーション

# Revit SDK

- Revit Developer Center で無償ダウンロード可能

[www.autodesk.co.jp/developer-network/platform-technologies/revit](http://www.autodesk.co.jp/developer-network/platform-technologies/revit)

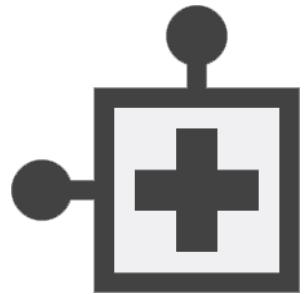
- 多数のサンプルプロジェクト (Revit SDK 2019 : 146 種類)
  - RevitAPIDllsPathUpdater.exe で全プロジェクトの参照設定を解決



- Add-In Manager : *Revit 2019 SDK/Add-In Manager* フォルダ

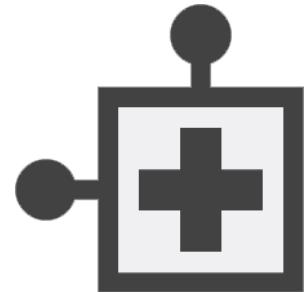


# アドイン マニフェストとは



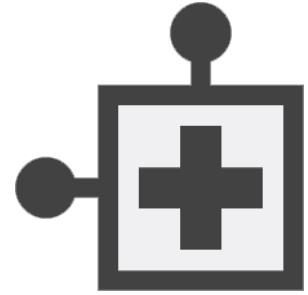
- Revit 起動時にアドイン特性を認識、ロードさせる役割
- マニフェストは拡張子 .addin を持つ XML ファイル
- XML タグで情報を指定
- 代表的なタグ
  - **<Addin Type>** : 外部コマンドまたは外部アプリケーション
  - **<Text>** : [アドイン] タブの [外部ツール] パネルに表示
  - **<FullClassName>** :  
名前空間を含む完全なクラス名 (大文字小文字を区別)
  - **<Assembly>** : アセンブリファイル (.dll) へのパス
  - **<ClientId>** : コマンド/アプリケーションを識別する GUID  
**<AddInId>** タグ相当

# アドイン マニフェスト : 外部コマンド例



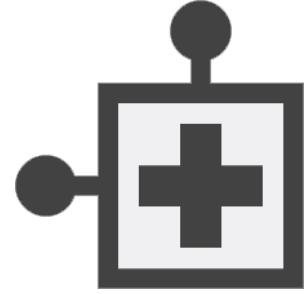
```
<RevitAddIns>
  <AddIn Type="Command">
    <Text>Hello World</Text>
    <Description>External Command for Hello World</Description>
    <Assembly>RevitAddin1.dll</Assembly>
    <FullClassName>RevitAddin1.Command</FullClassName>
    <ClientId>205693ad-2e34-4462-a05f-754a981ae69b</ClientId>
    <VendorId>technology_perspective</VendorId>
    <VendorDescription>Perspective From Japan,
      https://adndevblog.typepad.com/technology_perspective</VendorDescription>
  </AddIn>
</RevitAddIns>
```

# アドイン マニフェスト : 外部アプリケーション例



```
<RevitAddIns>
  <AddIn Type="Application">
    <Name>Hello World</Name>
    <Assembly>RevitAddin1.dll</Assembly>
    <FullClassName>RevitAddin1.App</FullClassName>
    <ClientId>1c6c3c7d-3617-4821-96a6-989755493253</ClientId>
    <VendorId>technology_perspective</VendorId>
    <VendorDescription>Perspective From Japan,
https://adndevblog.typepad.com/technology\_perspective</VendorDescription>
  </AddIn>
</RevitAddIns>
```

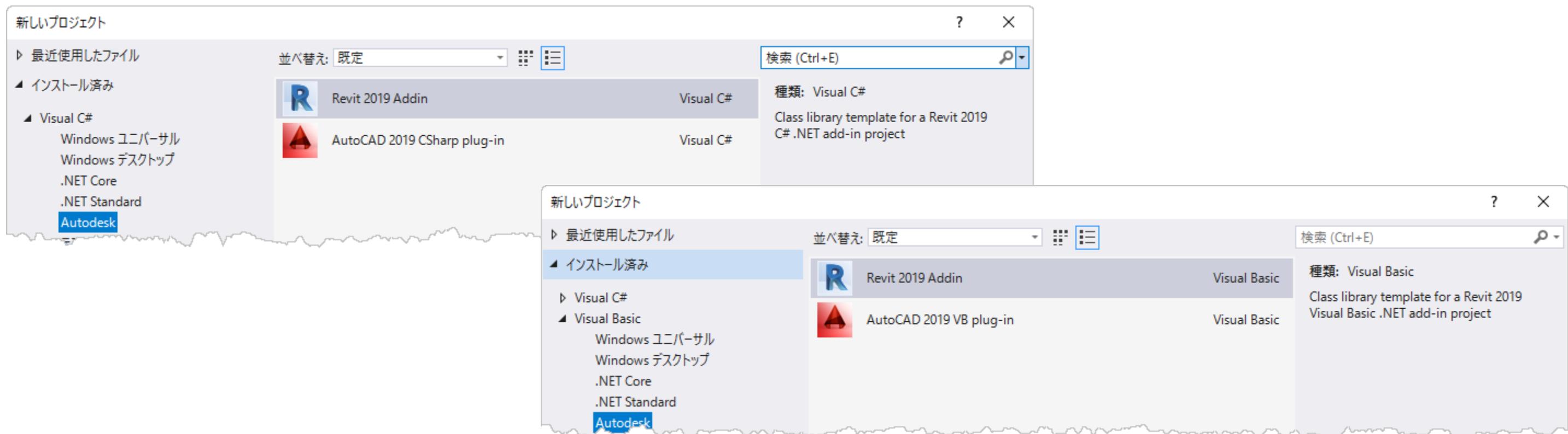
# アドイン マニフェストの配置位置



- 全 Windows ログインユーザ向け  
¥ProgramData¥Autodesk¥Revit¥Addins¥2019
- Windows ログインユーザ毎  
¥Users¥<UserName>¥AppData¥Roaming¥Autodesk¥Revit¥Addins¥2019
- 通常アドイン本体 (.dll ファイル) と同じパスに配置
- 参考：App ストアで入手したアドイン  AUTODESK® APP STORE
  - ¥ProgramData¥Autodesk¥ApplicationPlugins
  - PackageContents.xml ファイルと共にバンドルとして配置される

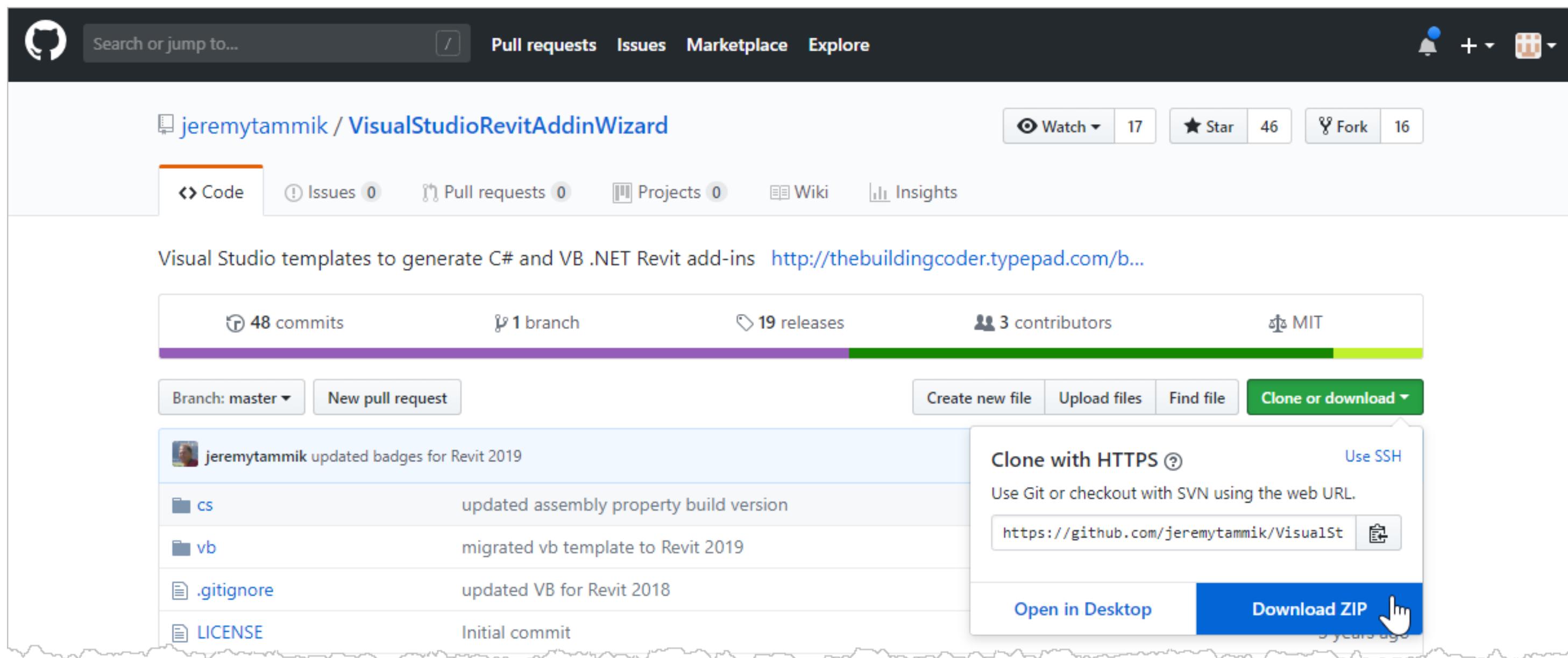
# Revit Add-In Wizard (アドイン ウィザード)

- Revit アドイン用スケルトンプロジェクトを作成
  - マニフェストも自動生成、配置
  - Visual Studio 用の C#、VB.NET テンプレート
  - GitHub 上で公開済 (インストーラ無)
  - ダウンロードして手動でインストール (推奨)



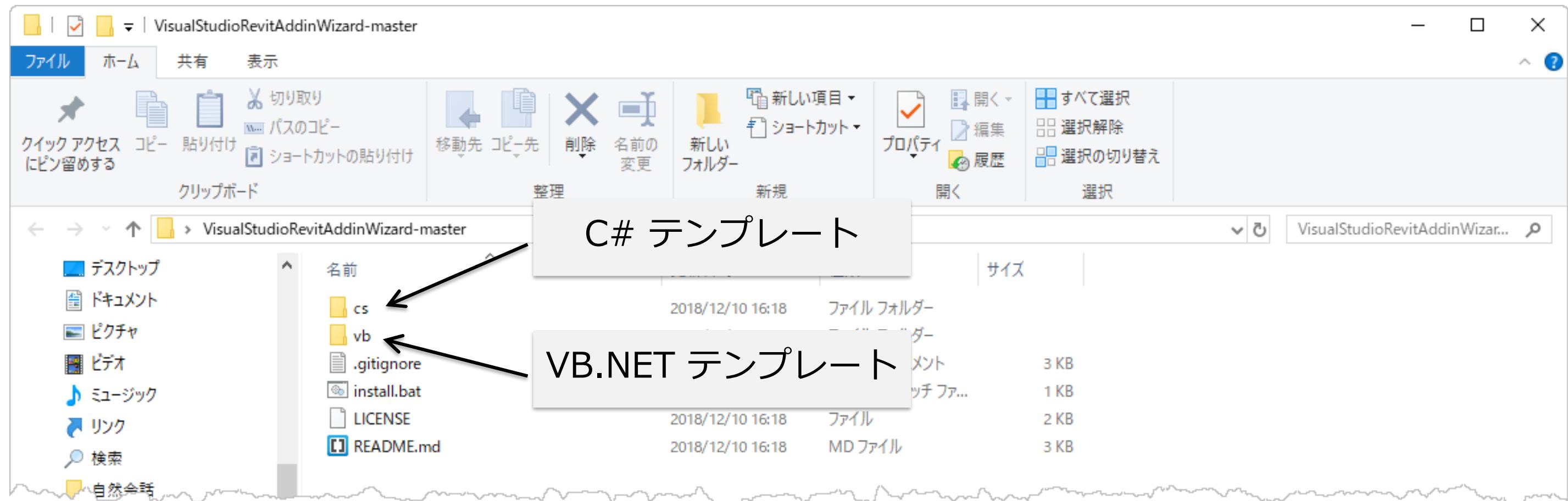
# Revit Add-In Wizard のセットアップ

1. <https://github.com/jeremytammik/VisualStudioRevitAddinWizard> から *VisualStudioRevitAddinWizard-master.zip* ファイルをダウンロード



# Revit Add-In Wizard のセットアップ～続き

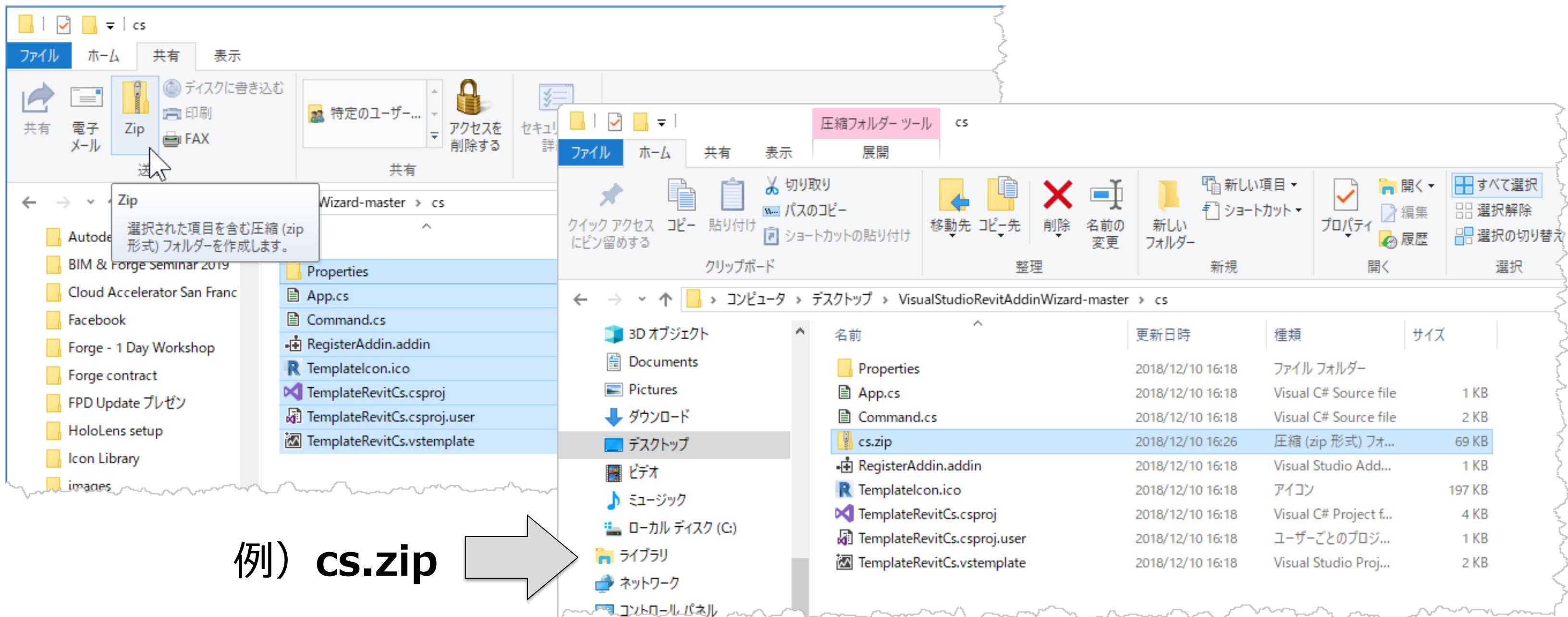
## 2. ダウンロードした ZIP を解凍



- Install.bat は実行に管理者権限の設定があり面倒

# Revit Add-In Wizard のセットアップ～続き

## 3. cs、vb フォルダへ移動して、それぞれ中身を任意名で ZIP 圧縮



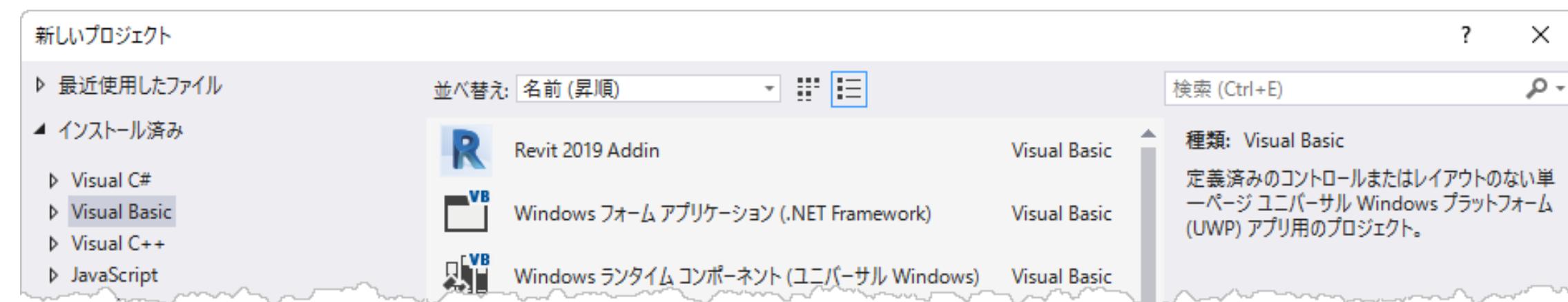
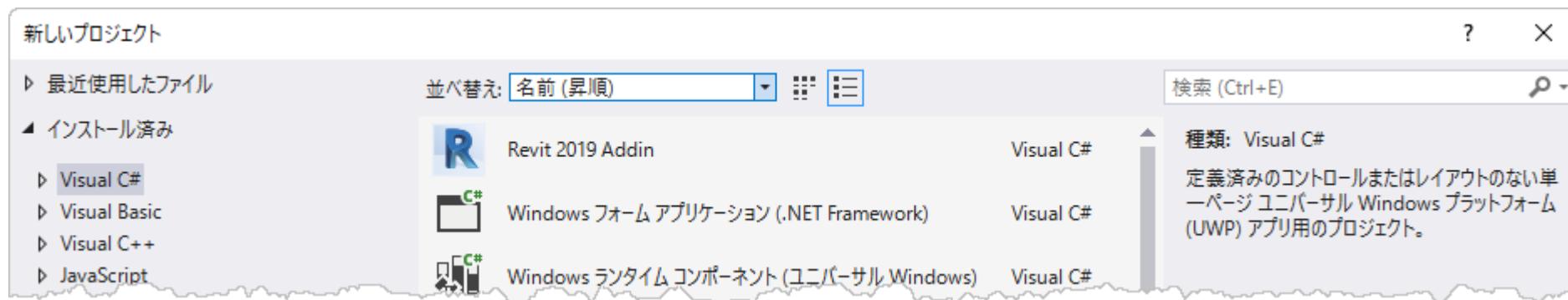
# Revit Add-In Wizard のセットアップ～続き

## 4. 圧縮した cs.zip、vb.zip を Visual Studio の ProjectTemplates フォルダ下の言語別フォルダへそれぞれ移動

- C#：
  - C:\Users\<ログインユーザ名>\OneDrive\Documents\Visual Studio 2017\Templates\ProjectTemplates\Visual C#  
または  
C:\Users\<ログインユーザ名>\Documents\Visual Studio 2017\Templates\ProjectTemplates\Visual C#
- VB.NET：
  - C:\Users\<ログインユーザ名>\OneDrive\Documents\Visual Studio 2017\Templates\ProjectTemplates\Visual Basic  
または  
C:\Users\<ログインユーザ名>\Documents\Visual Studio 2017\Templates\ProjectTemplates\Visual Basic

# Revit Add-In Wizard のセットアップ～続き

## 5. Visual Studio を起動して確認！



# 参考：DB レベル外部アプリケーションの実装（C# 例）

```
class App : IExternalDBApplication
{
    public Result OnStartup(ControlledApplication application)
    {
        return ExternalDBApplicationResult.Succeeded;
    }

    public Result OnShutdown(ControlledApplication application)
    {
        return ExternalDBApplicationResult.Succeeded;
    }
}
```

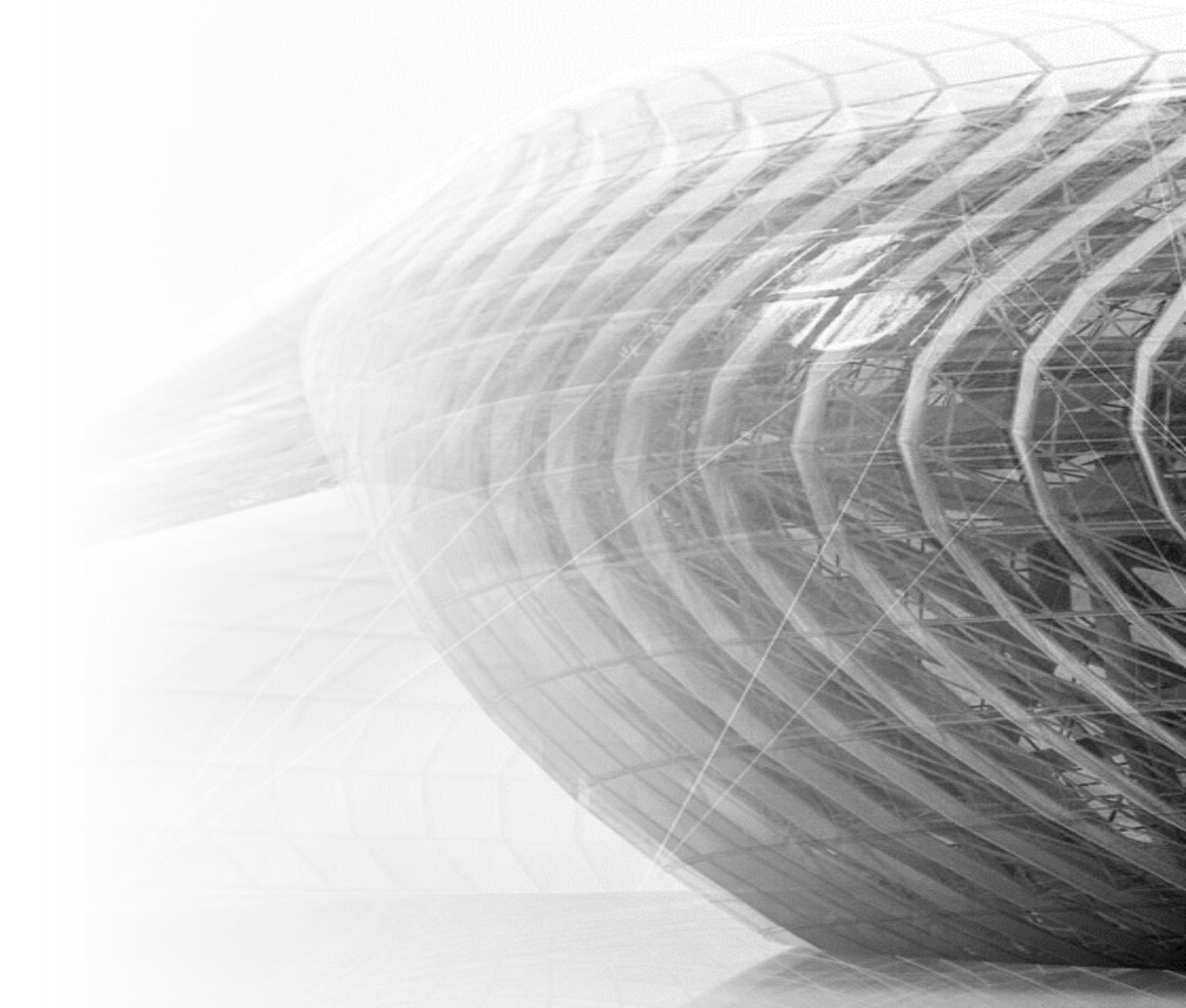
IExternalDBApplication からのクラス派  
生

← Revit の起動時に OnStartup が呼び出される

← Revit の終了時に OnShutdown が呼び出される

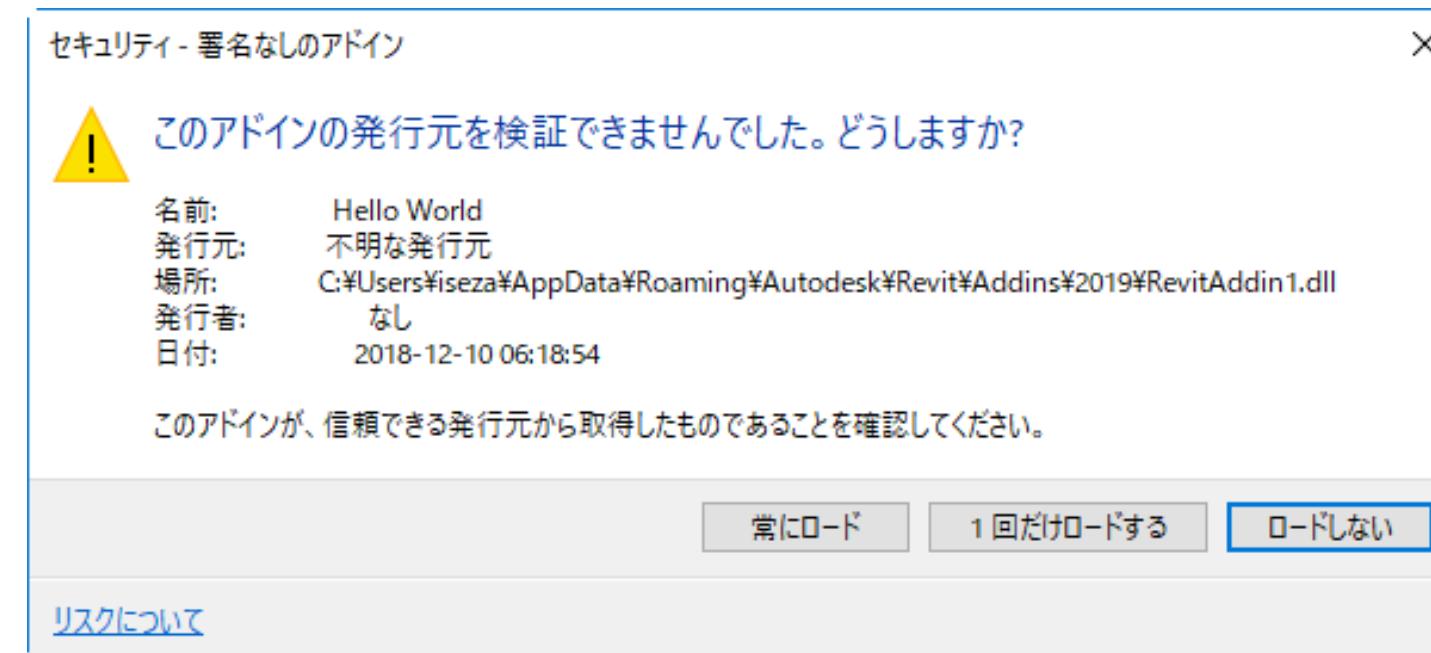


**補足**



# アドインのデジタル署名

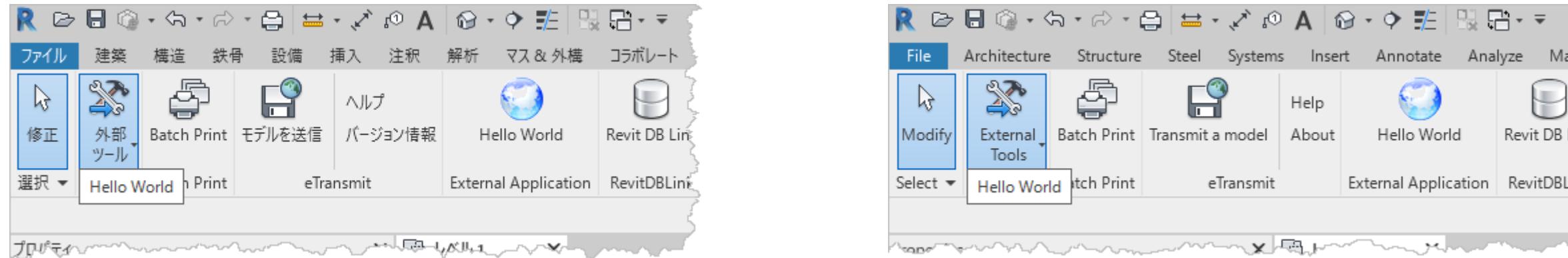
- Revit はアドインロード時にデジタル署名をチェック
  - アドイン開発/提供者を明確にして信頼性を向上
  - オートデスクはアドインへのデジタル署名を推奨！
  - デジタル署名のないアドイン ロード時：



- 詳細は Revit 開発者用ガイド参照

# 多言語 Revit 環境でのアドイン

- アドイン自体に言語選択はありません



- 参考：「Revit を他の言語で使用する」
  - <https://knowledge.autodesk.com/ja/support/revit-products/troubleshooting/caas/CloudHelp/clouddhelp/2017/JPN/Revit-Installation/files/GUID-BD09C1B4-5520-475D-BE7E-773642EEBD6C-.htm.html>
  - マニフェストを使ったアドインリソースの切り替えは可能
    - <LanguageType> タグとリソース DLL の併用
- Revit プロジェクト依存データに注意！
  - 海外で作成されたファミリ名、カテゴリ名など

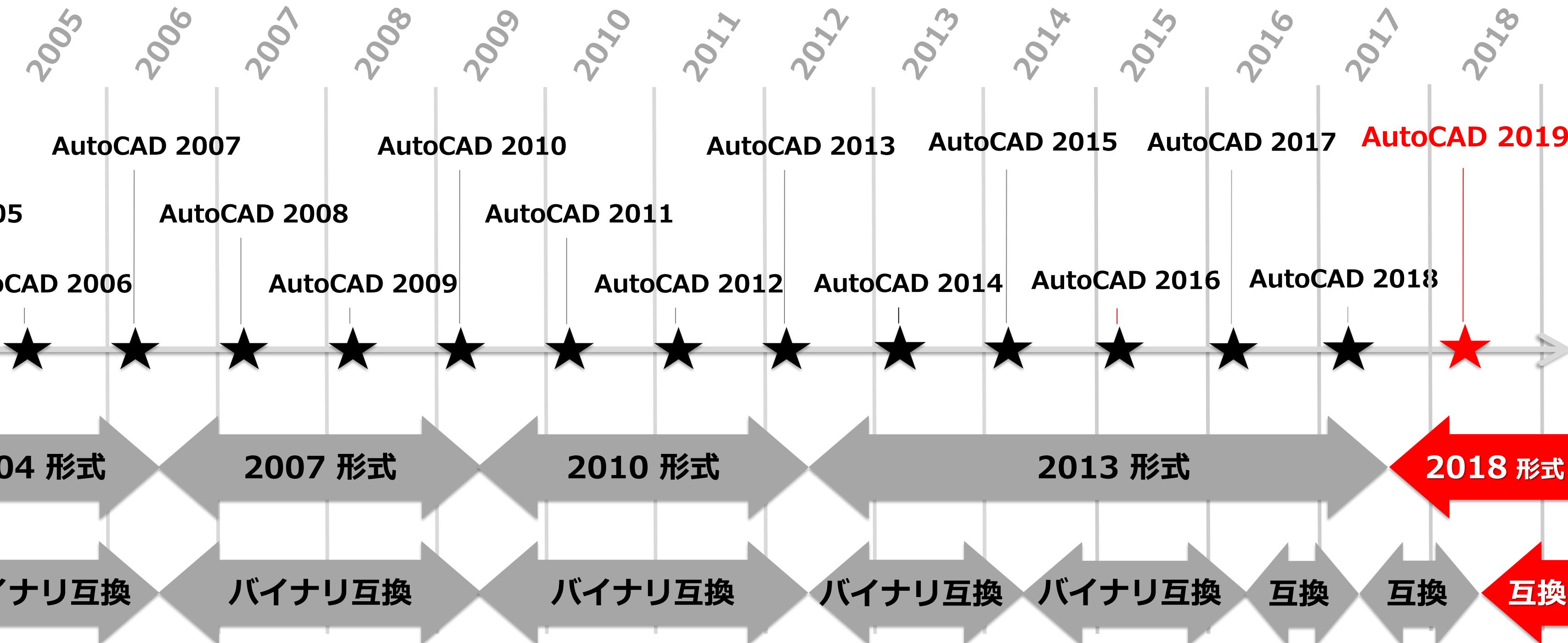
# Revit バージョン間の互換性

- Revit プロジェクト (.rvt) 、ファミリ (.rfa)
  - Revit によるアップグレードが必須
- Revit API
  - バージョン毎にクラス、メンバ関数の**統廃合**、**変更**が発生
  - ソースコードを一部修正、再ビルド、テストが必須
  - Revit が利用する .NET Framework との同期も必要
- AutoCAD のようなバージョン間互換維持の考え方はない
  - Revit データベース (ファイル内構造) の改良、改善
  - 新機能、新テクノロジの積極採用



# 参考：AutoCAD の場合

## 図面と API カスタマイズの互換性



# デスクトップ製品 SLA で禁止されている行為

<https://www.autodesk.com/company/terms-of-use/jp/general-terms>

## 15.3 提供物の利用の許容範囲

お客様は、全ての適用法に従ってのみ提供物のアクセス及び利用（並びにアクセス及び利用の許可）を行います（かつ全ての当該適用法に従います）。本規約（追加規約若しくは特別規約を含む）で明示的に許可された場合を除き、又はオートデスクが書面で明示的に別段の許可を行った場合を除き、お客様は、以下の行為を行いません。

- 提供物の全て又は一部の複製、改変、翻案、翻訳、移植、又はこれの二次的著作物の作成、ただし、反対の趣旨の契約上の禁止に関わらず、適用法で明示的に許可された場合を除く。
- 提供物（提供物の機能を含む）の全て若しくは一部の第三者へのサブライセンスの許諾、配布、送信、販売、賃貸、貸付、若しくはその他の方法で利用可能とすること、又は（サービスビューロベースその他による）第三者への提供物の機能の提供
- インターネット、広域ネットワーク（WAN）、その他のローカルでないネットワーク、仮想プライベートネットワーク（VPN）、アプリケーション仮想化技術、リモート仮想化技術、ウェブホスティング、タイムシェアリング、サービスとしてのソフトウェア、サービスとしてのプラットフォーム、サービスとしてのインフラ、クラウドその他のウェブベース、ホスト型等のサービス上又はこれらを通じての提供物のアクセス若しくは利用（オートデスクによるインターネットを通じた提供を除く）

さらに、お客様は以下の行為を行いません。

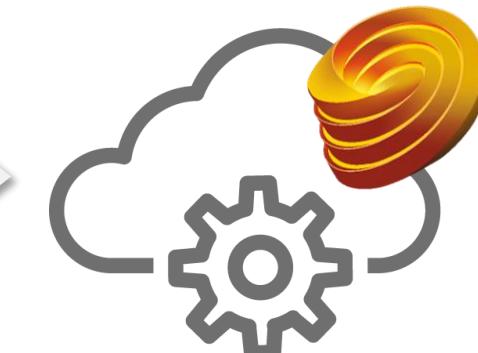
<中略>

- 提供物を「遠隔コード」のための記憶装置として又はその他のウェブページ若しくはインターネットリソースへの「入り口」若しくは「道標」として使用すること（提供物が提供されるサイト内であるか又はサイトの域内を越えるかを問わない）

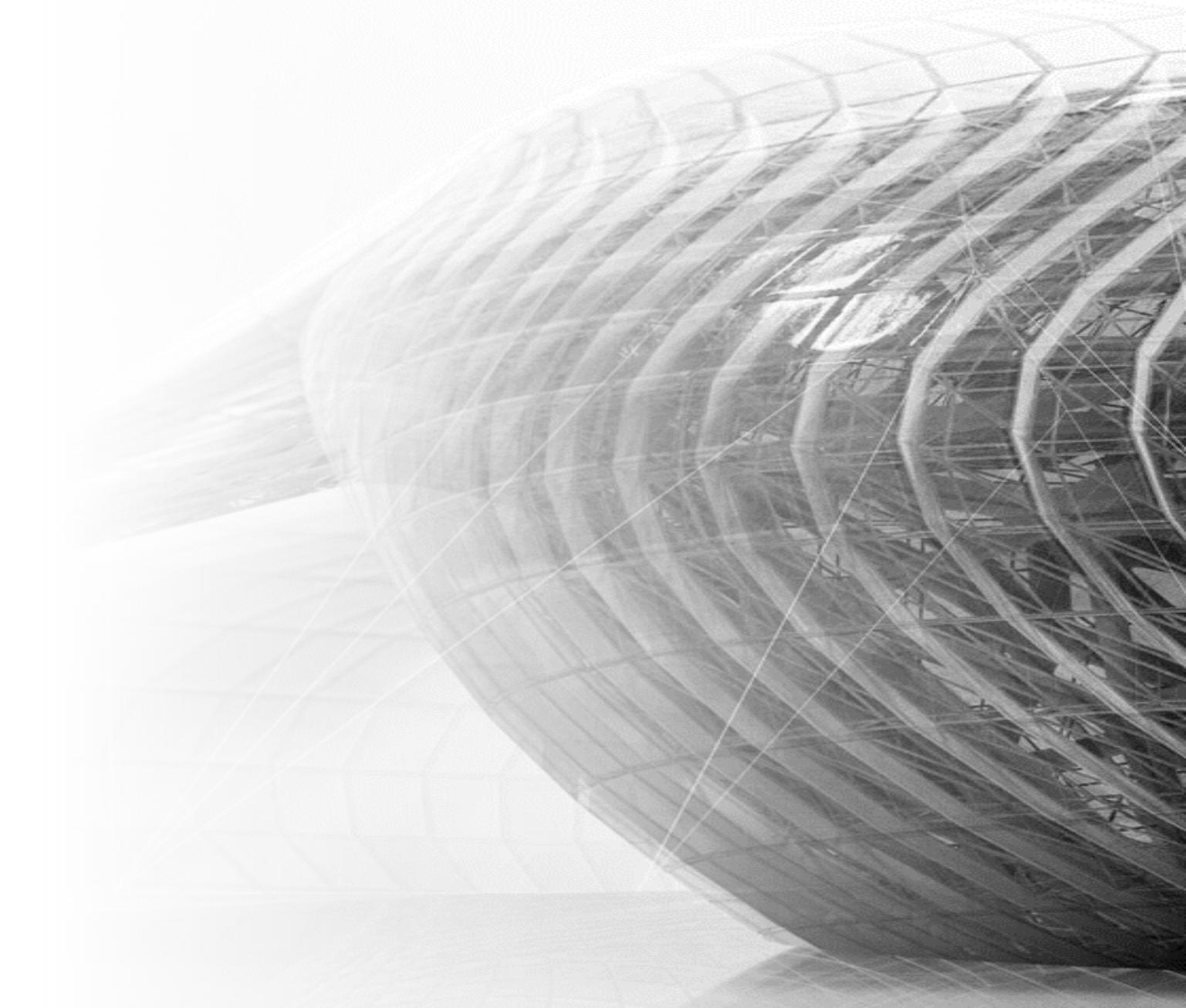
<中略>



共有サーバー/クラウドに独自にホストした  
Revitを使った不特定多数へのサービス提供は  
**Design Automation API for Revit** で



# ご案内





# BIM & Forge セミナー 2019



- **期日&場所：**
  - **1月22日(火) 東京 - 13:00~17:20 (予定)**
    - 場所：東京セミナー/トレーニングルーム (定員：100名)
  - **1月25日(金) 大阪 - 13:00~17:20 (予定)**
    - 場所：大阪セミナールーム (定員：40名)
- **参加費：無償**
- **対象：**BIM 360による効率化、BIM用途でForgeに興味のある方
- **詳細：**

[https://adndevblog.typepad.com/technology\\_perspective/2018/12/announcing-bim-and-forge-seminar.html](https://adndevblog.typepad.com/technology_perspective/2018/12/announcing-bim-and-forge-seminar.html)

(Revit API&BIM セミナー後に公開予定)



# BIM & Forge セミナー 2019



## ■ アジェンダ

13:00～14:00 BIM 360 ファミリ製品とデータ管理

14:00～14:10 <休憩>

14:10～15:00 Forge 概要と BIM 360 の関係

15:00～15:15 <休憩>

15:15～16:05 Forge による BIM 360 Docs 統合

16:05～16:20 <休憩>

16:20～17:20 Forge Design Automation API for Revit での自動化



AUTODESK®

Make anything.

Autodesk and the Autodesk logo are registered trademarks or trademarks 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.

© 2017 Autodesk. All rights reserved.