

SD473594

Design Automation for Revit 基礎から応用へ

小笠原 龍司
オートデスク株式会社

学習目標

- Revit アドインの処理をクラウドで実行できるように自動化する方法を学習します。
- Design Automation for Revit を使用する Web アプリケーションの開発方法を学習します。
- サンプルを通じて Forge アプリケーション開発を促進します。
- アダプティブ コンポーネントを使用してコンセプトデザインを生成する方法を学習します。
- 遺伝的アルゴリズムを使用して最適なファミリのレイアウトを生成する方法を学習します。

説明

オートデスクでは、コラボレーションの促進と大規模なオートメーションを実現するために、お客様の業務をクラウドに移行する必要性を強く認識しています。このクラスは、Revit アドイン開発者が Design Automation for Revit を使用することを想定し、基礎的な解説に加え、より応用的な内容を取り上げます。

5 つのケーススタディについてデモを交えてご紹介し、それらを実装するために必要な手順とコードを説明します。アダプティブ コンポーネントを使用してコンセプトデザインを生成する方法を学習します。遺伝的アルゴリズムを使用して最適なファミリのレイアウトを生成する方法を学習します。複数の Revit バージョンをサポートするために Activity と Appbundle を管理する方法を紹介します。エンドツーエンドで動作するコードサンプルへのリンクを共有し、参加者ご自身で拡張できるように致します。Revit ワークフローをサポートするクラウドアプリケーションの構築方法について、確かな洞察を得ることができます。

スピーカーについて

2014 年からオートデスクにて、Revit や Navisworks 等、デスクトップ製品 API・アドイン開発・API のサポート、及び Forge プラットフォーム API のエバンジェリストとして活動中。

前職では、Web アプリケーションエンジニアとして環境制御システムやリアルタイムコミュニケーションシステム、統合基幹業務システムの開発に従事。

クラスの概要

このクラスは、Design Automation for Revit の概要から始まり、過去にもご説明した内容を要約します。オートデスクのスピーカーによる以前のクラスへのリンクを次に示します。

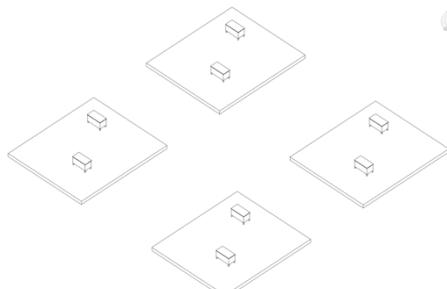
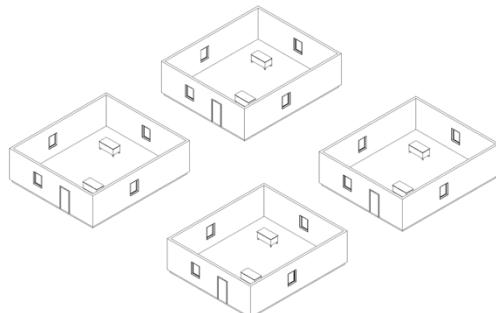
[SD323658: Getting Started on Design Automation for Revit on Forge](#)

[SD125457: Design Automation: Building Web Applications with Revit Data on Forge](#)

このクラスでは、チュートリアルサンプルを使用して Design Automation アプリケーションの技術的な詳細を説明します。まずは基本的な解説から始め、Design Automation サービス内で利用できる様々なオプションを理解するために、徐々に深く掘り下げていきます。現在、Design Automation アプリケーションは、AppBundle (別名アドイン) 内からインターネットにアクセスすることはできません。このクラスは、結果を生成してアップロードするために必要なデータを提供する方法を示します。

Delete Walls

基本的なコードサンプルの一つである "Hello World" アプリケーションです。このセクションでは、Activity 定義の詳細を参照して、Revit ファイルを入力として取得し、それを変更してクラウドストレージにアップロードするアプリケーションを構築する方法を確認します。このセクションでは、クラウド マシン上で実行されるコマンドラインの入力や、内部で使用する変数の詳細について説明します。AppBundle と Activity の関係と定義、Design Automation サービスがダウンロード・アップロードするためのファイル名の定義の方法を示します。



基本的な入力データセットを使用して、ワークフローを理解します。データセットとソースコードは、追加のマテリアルセクションに記載されています。Delete Walls サンプルは、Postman コレクションや、curl ベースの REST API ドキュメントなどの言語に依存しないドキュメントを含め、Forge チュートリアルの多くで使用されています。

Count It

このセクションでは、**Revit** ファイルを入力として取得し、そこから情報を抽出する別の基本的なサンプルを解説します。また、**Forge** アプリケーションから、ジョブにパラメーターを渡して、**AppBundle** で、パラメータに応じて条件分岐処理を実装する方法を示します。**Forge** アプリケーションから **JSON** データを **WorkItem** の引数にして **AppBundle** に送信し、**Newtonsoft** ライブラリ（**JSON** データをシリアル化・デシリアル化するための **nuget** ライブラリ）で利用する方法を学びます。

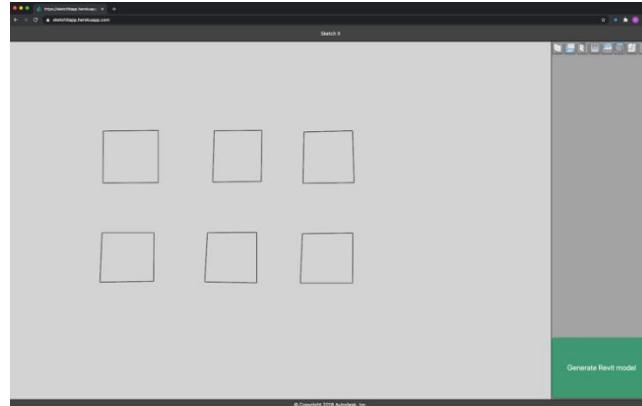
```
{
  "walls": 16,
  "floors": 4,
  "doors": 4,
  "windows": 16,
  "total": 40
}
```

このアプリケーションは理解が簡単で、入力されたカテゴリの要素数をカウントする非常に基本的な **Revit API** のルーチンを使用します。**Revit** ファイルを入力し、**Revit** ファイルではないファイルを生成して出力する方法について学習します。

これは、**Revit** モデルからのデータ抽出を構築しようとしている人に最適なサンプルです。このアクティビティ定義を使用して、複雑なワークフローの **AppBundle** に置き換えて簡単に利用できます。

Sketch It

Sketch It アプリケーションは、エンドツーエンドの **Web** アプリケーションを理解するシンプルなサンプルです。このアプリケーションでは、ユーザが **Web** ブラウザ上に壁や床を描画し、ブラウザ上に描画された線に対応する壁や床を持つ **Revit** ファイルを生成することができます。このアプリケーションは、背後で、ブラウザ上に描画された点/線の座標データを **Design Automation** に送信し、ユーザーが **Forge Viewer** で結果を表示できるように変換します。



このサンプルでは、このアプリケーションを構築するために必要な **Design Automation** の **Activity** と、**AppBundle**、及び **WorkItem** を実装する方法を学習します。前のセクションで解説した **json** のシリアル化方法をベースに、このセクションでは、より複雑で大規模な **json** データの入力のワークフローについて説明します。前のセクションとは異なり、この例では入力する **Revit** ファイルが存在しないため、**Revit** ファイルを新規に作成するワークフローも扱います。

アプリケーション: <https://sketchitapp.herokuapp.com/>

ソース コード: <https://github.com/Autodesk-Forge/forge-sketchit-revit>

Stadium Panel

このセクションでは、Revit API と Design Automation の機能を組み合わせて、アダプティブコンポーネントとファミリパラメータを使用してコンセプトデザインを生成する方法について説明します。アダプティブコンポーネントは点ベースのファミリで、アダプティブポイントを使用してファミリをパラメータ化できます。ファミリインスタンスパラメータと同様に、アダプティブポイントはファミリインスタンスレベルで変更して、フレキシブルにジオメトリを生成し、ファミリ定義を再利用することができます。



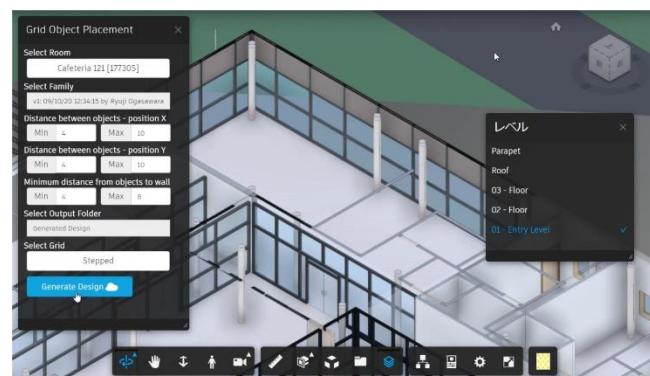
Design Automation for Revit は、デザインを前もって視覚化することが難しい場合に有効で、コンセプトデザインファイルの生成を自動化するプラットフォームを提供し、それらをすべてブラウザで表示してから、選択するか、さらに微調整するかを決定できます。このセクションでは、パネルファミリを選択し、ランダムにファミリのインスタンスパラメータを変更して、複雑で美しい構造を生成します。2つのファミリファイルと、200行未満のコードで、複雑で美しい構造を出力します。

Grid Object Placement

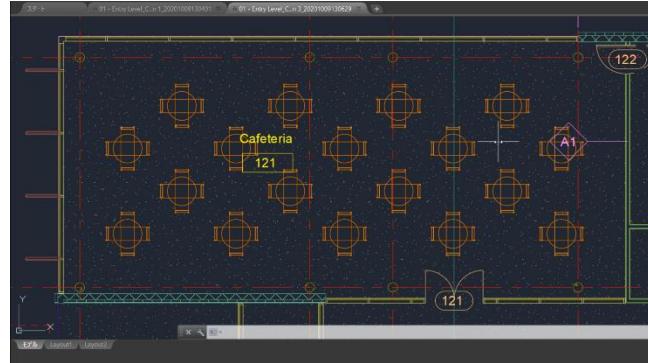
このセクションでは、グリッドパターンを使用して部屋に家具ファミリを配置して、その平面図を DWG ファイルとして書き出す方法をご紹介します。

グリッドパターンは、長方形かひし形を選ぶことができ、X 軸と Y 軸の距離を、指定した範囲内のランダム値で作成し、遺伝的アルゴリズムの仕組みを利用して、世代ごとに最も多くのファミリを配置できたレイアウトを書き出します。

最終的に世代ごとの結果の DWG ファイルを複数まとめて ZIP 圧縮して BIM 360 Docs にアップロードします。



このサンプルでは、遺伝的アルゴリズムのオープンソースライブラリを使用しており、Revit 2021 のジェネレーティブデザインと遺伝的アルゴリズムのコンセプトについて解説します。またソースコードを通じて、Design Automation の機能を Forge Viewer Extension として作成する方法、BIM 360 のデータを Design Automation で利用する方法も確認できます。



クラスの参考 URL

ソース コード: <https://github.com/rahulbhobe/autodesk-university-2020-sd473692>