



Forge Online

AutoCAD タスクの自動化： Design Automation API for AutoCAD の理解

伊勢崎俊明

オートデスク 株式会社



FORGE
DevCon
2016

AutoCAD I/O
(Beta)



Design Automation API
v1



Authentication API



Model Derivative API



Data Management API



Viewer



Reality Capture API
(Beta)



BIM 360 API
(Beta)



WebHooks API
(Beta)



WebHooks API



Design Automation API
v3



View and Data API
(Beta)

FORGE

ReCap Photo API
(Beta)



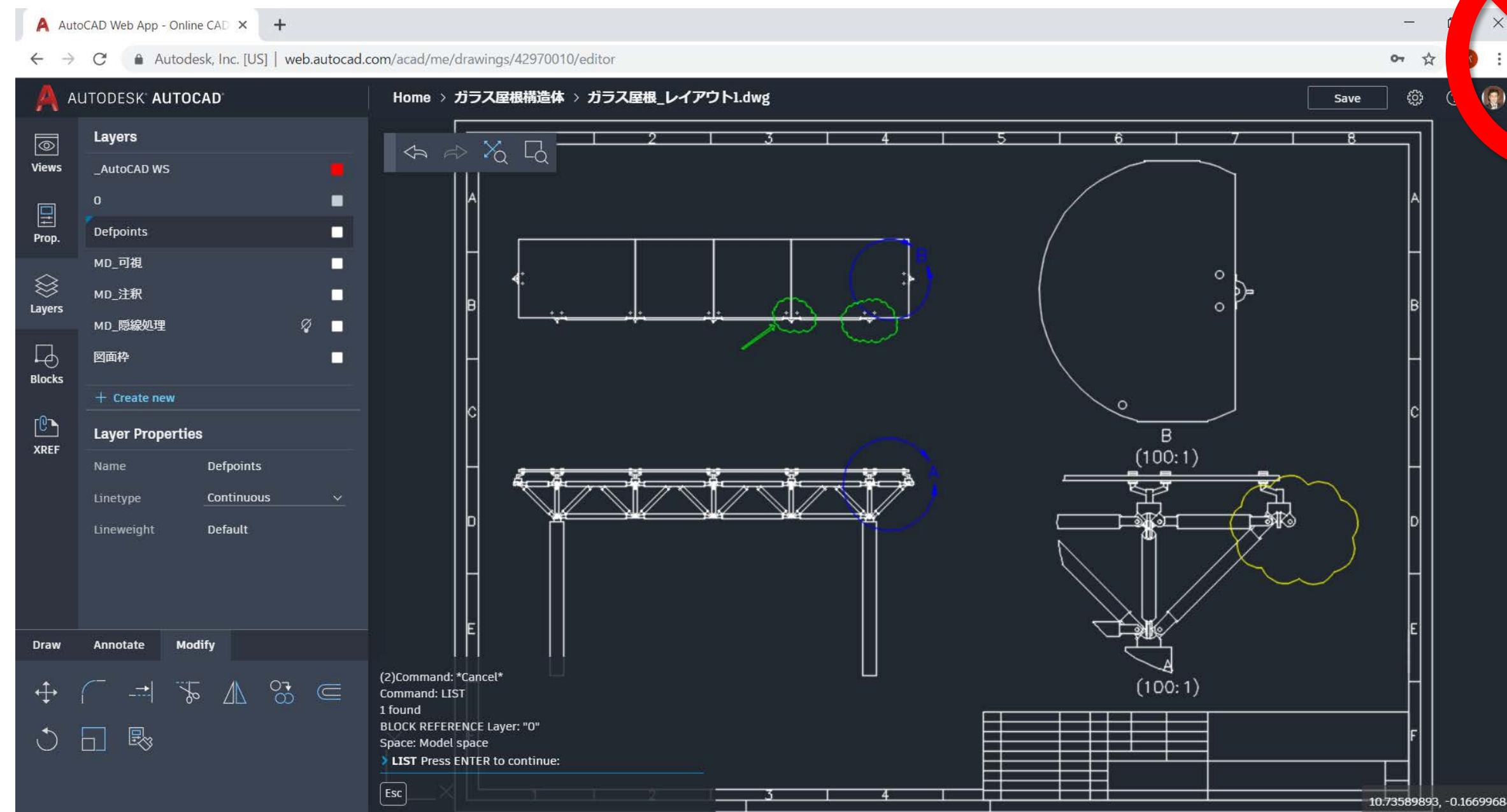
Reality Capture API



BIM 360 API

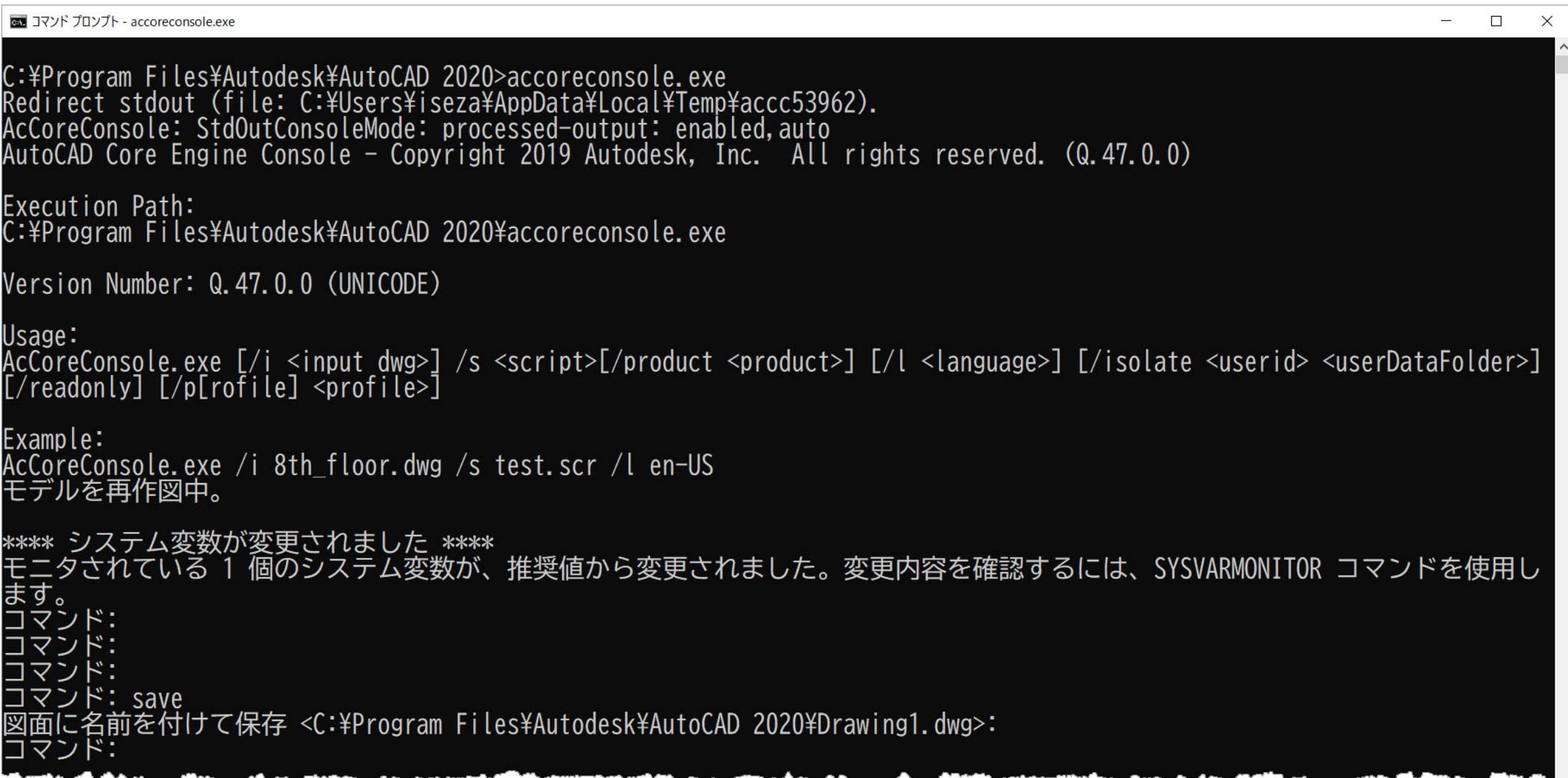
Design Automation API の間違った理解

- AutoCAD Web のアドイン開発環境ではありません



AutoCAD コアエンジン

- AutoCAD に同梱されているコマンド プロンプト版 AutoCAD
- **AcCoreConsole.exe** : リボン、ツールレバーなどの GUI を省略



```
C:\Program Files\Autodesk\AutoCAD 2020>accoreconsole.exe
Redirect stdout (file: C:\Users\iseza\AppData\Local\Temp\accc53962).
AcCoreConsole: StdOutConsoleMode: processed-output: enabled, auto
AutoCAD Core Engine Console - Copyright 2019 Autodesk, Inc. All rights reserved. (Q.47.0.0)

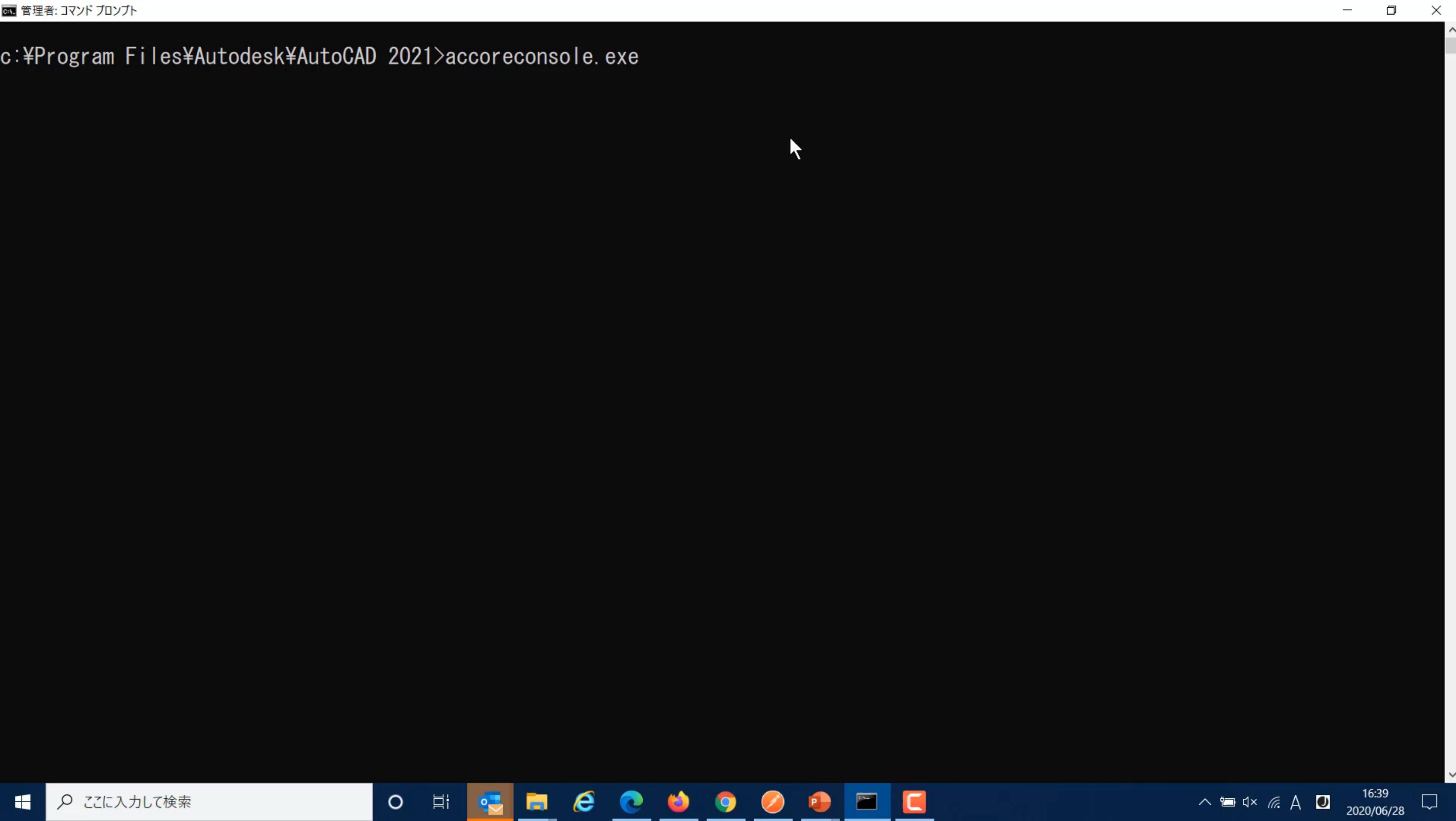
Execution Path:
C:\Program Files\Autodesk\AutoCAD 2020\accoreconsole.exe

Version Number: Q.47.0.0 (UNICODE)

Usage:
AcCoreConsole.exe [/i <input dwg>] /s <script>[/product <product>] [/l <language>] [/isolate <userid> <userDataFolder>]
[/readonly] [/p[rofile] <profile>]

Example:
AcCoreConsole.exe /i 8th_floor.dwg /s test.scr /l en-US
モデルを再作図中。

**** システム変数が変更されました ****
モニタされている 1 個のシステム変数が、推奨値から変更されました。変更内容を確認するには、SYSVARMONITOR コマンドを使用します。
コマンド:
コマンド:
コマンド:
コマンド:
コマンド: save
図面に名前を付けて保存 <C:\Program Files\Autodesk\AutoCAD 2020\Drawing1.dwg>:
コマンド:
```



AcCoreConsole.exe の能力

- GUI 実装の省略による省メモリと高速化を実現
- GUI を表示しないコマンド、カスタムコマンドの実行が可能
- アドイン (.NET API、ObjectARX、AutoLISP) のロード
- AutoCAD 同梱版は自動ローダーの機能はなし
- 起動時は新規図面をメモリ上に作成した状態
- 指定した図面をメモリ上に開くことが可能
- 作図処理後の図面を保存することも可能
- (ローカル環境でのテスト利用には管理者権限でのコマンドプロンプトの起動を推奨)

accoreconsole.exe と acad.exe の比較

- **accoreconsole.exe**
 - コマンド プロンプトから起動
 - UI のない AutoCAD
 - UI がないため動作が軽快
 - アドインのロードが可能
 - UI 表示のないアドイン限定
 - バージョン依存
 - 自動ローダーを認識しない
- **acad.exe**
 - ショートカットから起動
 - UI を持つ AutoCAD
 - 起動時等 UI 初期化に時間
 - アドインのロードが可能
 - UI 表示に関係なくロード可
 - バージョン依存
 - 自動ローダーを認識

DA4A で利用可能なコアエンジン バージョン

- デスクトップ製品と同期したコアエンジン バージョン
 - AppPackage はエンジン バージョンに合わせた作成が必須
 - コアエンジン ID の形式はコアエンジン毎に異なる
 - 2020年6月28日現在（最新+過去3バージョンがサポート対象）

“Autodesk.AutoCAD+20_1”	⇒	AutoCAD 2016
“Autodesk.AutoCAD+21”	⇒	AutoCAD 2017
“Autodesk.AutoCAD+22”	⇒	AutoCAD 2018
“Autodesk.AutoCAD+23”	⇒	AutoCAD 2019
“Autodesk.AutoCAD+23_1”	⇒	AutoCAD 2020
“Autodesk.AutoCAD+24”	⇒	AutoCAD 2021

利用手順と endpoint

1. Nickname の登録※

PATCH forgeapps/:id

2. AppPackage を登録 (アップロード パラメータ取得) ※ POST appbundles

3. AppPackage をアップロード※

POST dasprod-store.s3.amazonaws.com

4. AppPackage に Alias を設定※

POST appbundles/:id/aliases

5. Activity を登録

POST activities

6. Activity に Alias を設定

POST activities/:id/aliases

7. WorkItem を実行

POST workitems endpoint

8. WorkItem ステータスをチェック

GET workitems/:id



Forge Online

AutoCAD タスクの自動化： Activity のみの Design Automation

伊勢崎俊明
オートデスク 株式会社

DA4A PDF output

セキュリティ保護なし | forge-da4a-pdf-output.herokuapp.com

Upload

Download

Delete Activity

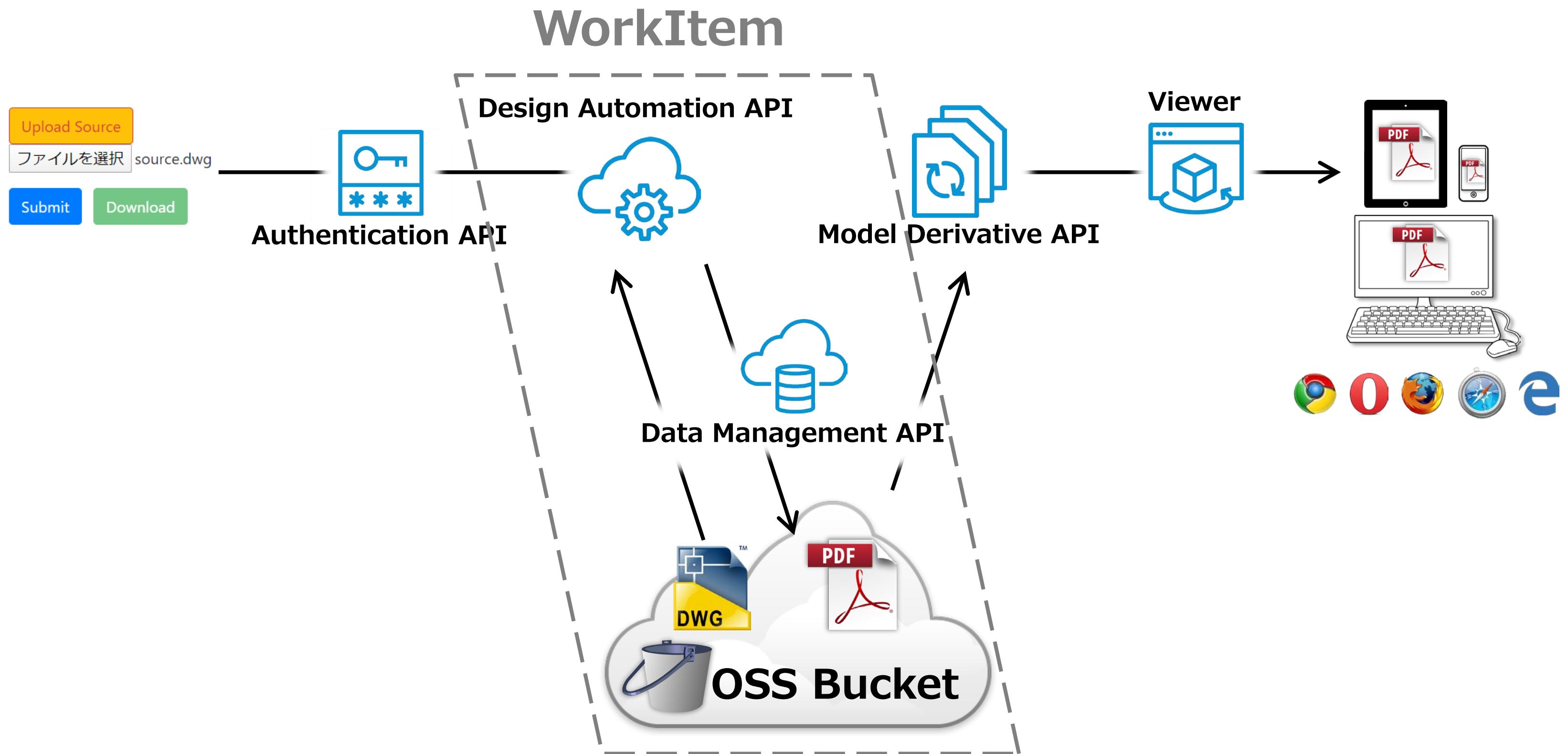
Register Activity

オートデスク 株式会社
東京オフィス
プロア プラン園
2001年5月24日
Gensler

'PDF Output' のワークフロー

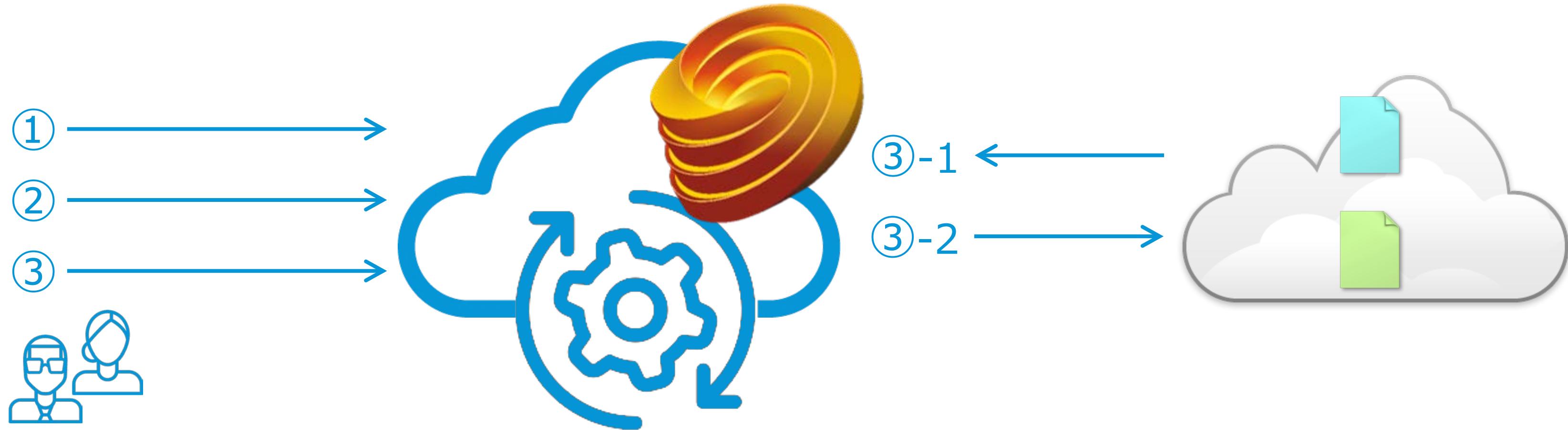
1. 'PDFplot' Activity と 'dev' Alias を事前に登録
2. ローカルから PDF 化する DWG を選択
 - OSS Bucket に選択した DWG をアップロード
3. 'PDFplot' WorkItem を登録（タスクを起動）
 - OSS Bucket から対象 DWG を作業領域にダウンロード
 - コアエンジン AcCoreConsole.exe が DWG をオープン
 - Activity の Script 記述に沿って作業領域に result.pdf を出力
 - 作業領域から result.pdf を OSS Bucket にアップロード
4. OSS Bucket の PDF を SVF 変換
 - Forge Viewer へ表示
5. OSS Bucket から result.pdf をローカルにダウンロード

'PDF Output' のワークフロー



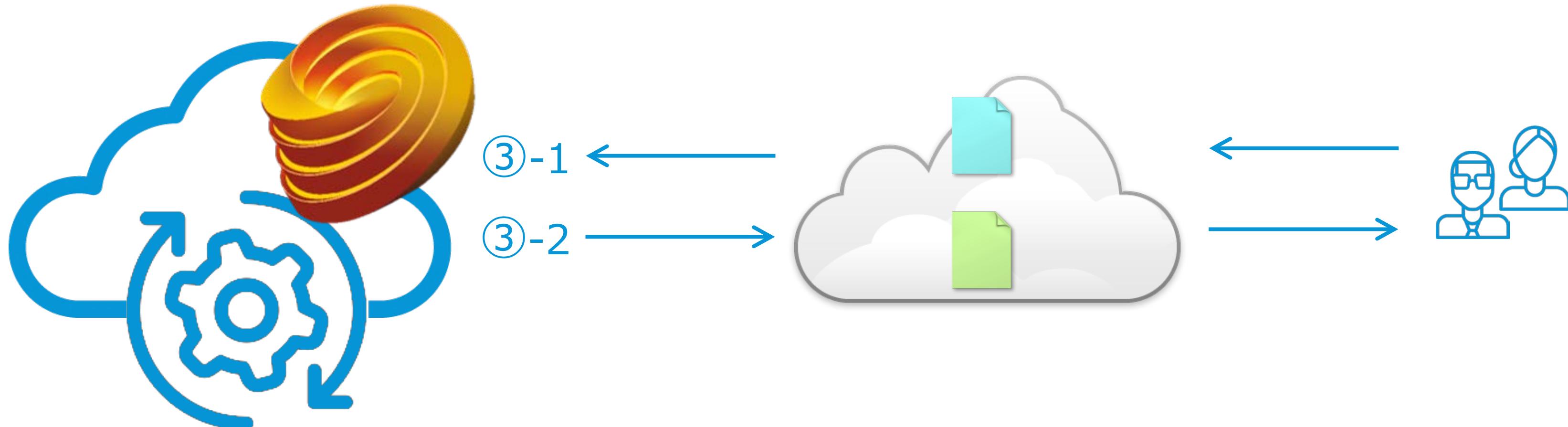
入力ファイルと出力ファイルの扱い

- プログラムでローカル PC のファイル操作の自動化は不可
 - ローカル PC から DA API 作業領域に参照ファイルを保存
 - DA API 作業領域から成果ファイルをローカル PC に保存
 - Web セキュリティ上の制限：ユーザ操作が必要



入出力ファイル指定は署名付き URL で

- リソースへのアクセス権を有効期限付きで一時的に提供
 - クラウドストレージから DA API 作業領域に参照ファイルを保存
 - DA API 作業領域から成果ファイルをクラウドストレージに保存
 - POST buckets/:bucketKey/objects/:objectName/signed



Activity (アクティビティ) 登録時の JSON 例

- パラメータ値の宣言

```
{  
    "id": "PDFPlot",  
    "commandLine": ["$(engine.path)\\accoreconsole.exe /i \"$(args[DWGInput].path)\" /s  
    \"$(settings[script].path)\""],  
    "parameters": {  
        "DWGInput": {  
            "zip": false,  
            "ondemand": false,  
            "verb": "get",  
            "description": "Source drawing",  
            "required": true  
        },  
        "PDFOutput": {  
            "zip": false,  
            "ondemand": false,  
            "verb": "put",  
            "description": "output PDF drawing",  
            "required": true,  
            "localName": "result.pdf"  
        }  
    },  
},  
}
```

Activity (アクティビティ) 登録時の JSON 例 ~ 続き

- パラメータ値の宣言

```
"settings": {  
    "script": {  
        "value": "_tilemode 0 -export _pdf _all result.pdf\n"  
    }  
},  
"engine": "Autodesk.AutoCAD+23_1",  
"appbundles": [],  
"description": "PDF output"  
}
```

- 実行コマンドは標準コマンド/システム変数の利用も可能
 - script 指定で記述
 - AppBundle の利用は必須ではありません
 - AppBundle 不使用の場合は [] で指定

WorkItem (ワークアイテム) 実行時の JSON

- パラメータ宣言に基づいて実際の値を指定

```
{  
  "activityId": "nqpwqsDLFGkS06LgA2mvaSXy5AeH5VSJ.PDFPlot+dev",  
  "arguments": {  
    "DWGInput": {  
      "url": "https://developer.api.autodesk.com/oss/v2/signedresources/xxxxx?region=US",  
      "headers": {  
        "Authorization": "Bearer xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",  
        "Content-type": "application/octet-stream"  
      },  
      "verb": "get"  
    },  
    "PDFOutput": {  
      "url": "https://developer.api.autodesk.com/oss/v2/signedresources/xxxxx?region=US",  
      "headers": {  
        "Authorization": "Bearer xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",  
        "Content-type": "application/octet-stream"  
      },  
      "verb": "put"  
    }  
  }  
}
```

補足：日本語 TrueType フォントについて

- DA4A稼働環境：
 - Amazon Machine Image でデプロイの英語 Windows 環境
 - 日本語フォントの扱い
 - 日本語シェイプフォントはコアエンジン付帯で利用可能
 - 日本語 TrueType フォントは一部利用可能
 - **MS ゴシック**、**MS P ゴシック**、**MS UI ゴシック**
 - **MS 明朝**、**MS P 明朝**、**游明朝**、**メイリオ**、**メイリオ UI**
 - 上記以外の利用：

AppBundle への同梱、または、
WorkItem 指定で実行時ダウンロードが必須

 - フォントが未解決の場合は FONTALT システム変数を採用

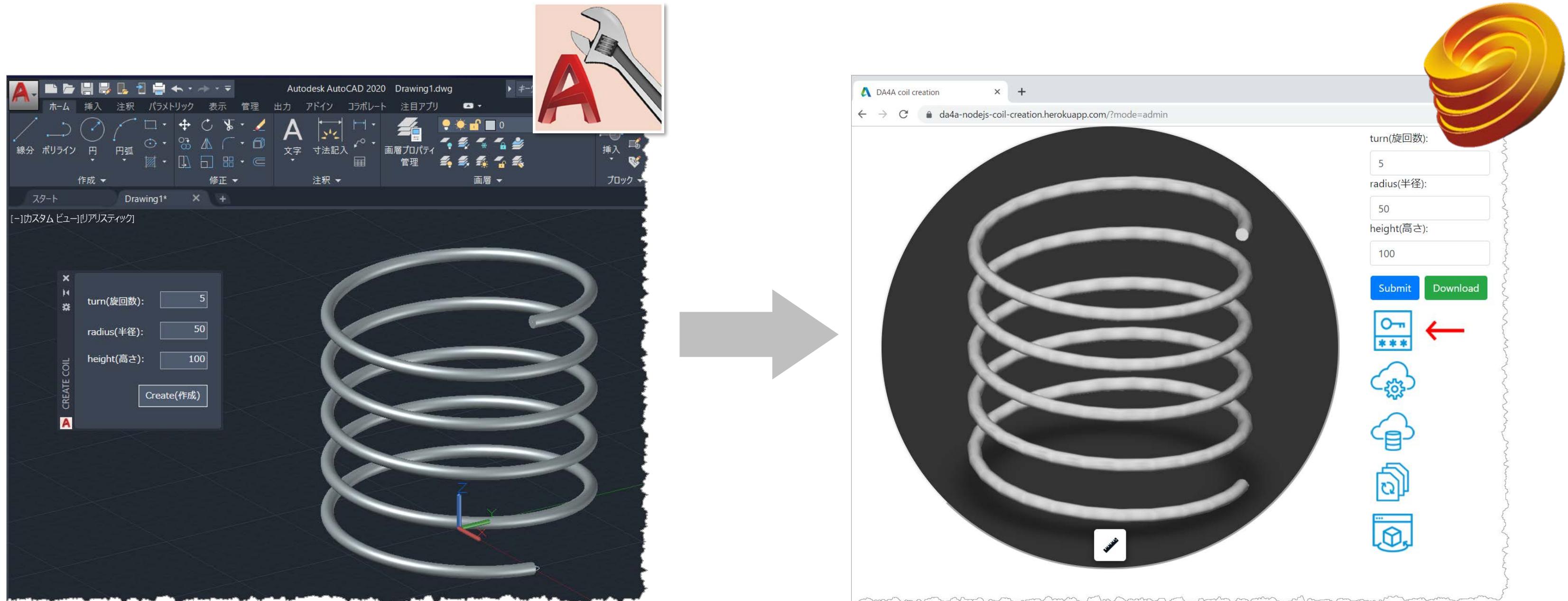


Forge Online

AutoCAD タスクの自動化： AppBundle での Design Automation

伊勢崎俊明
オートデスク 株式会社

CreateCoil アプリ

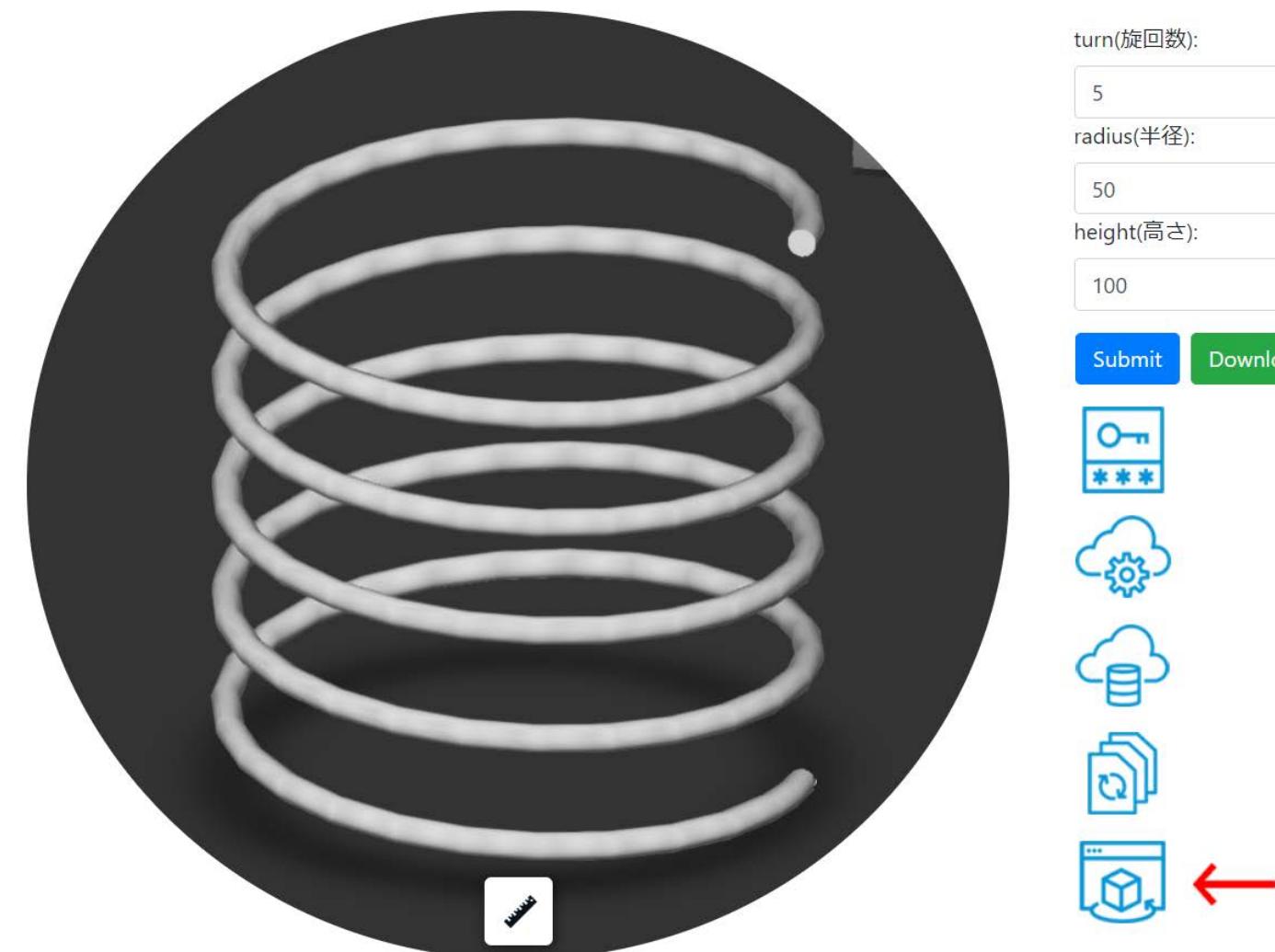


'CreateCoil' のワークフロー

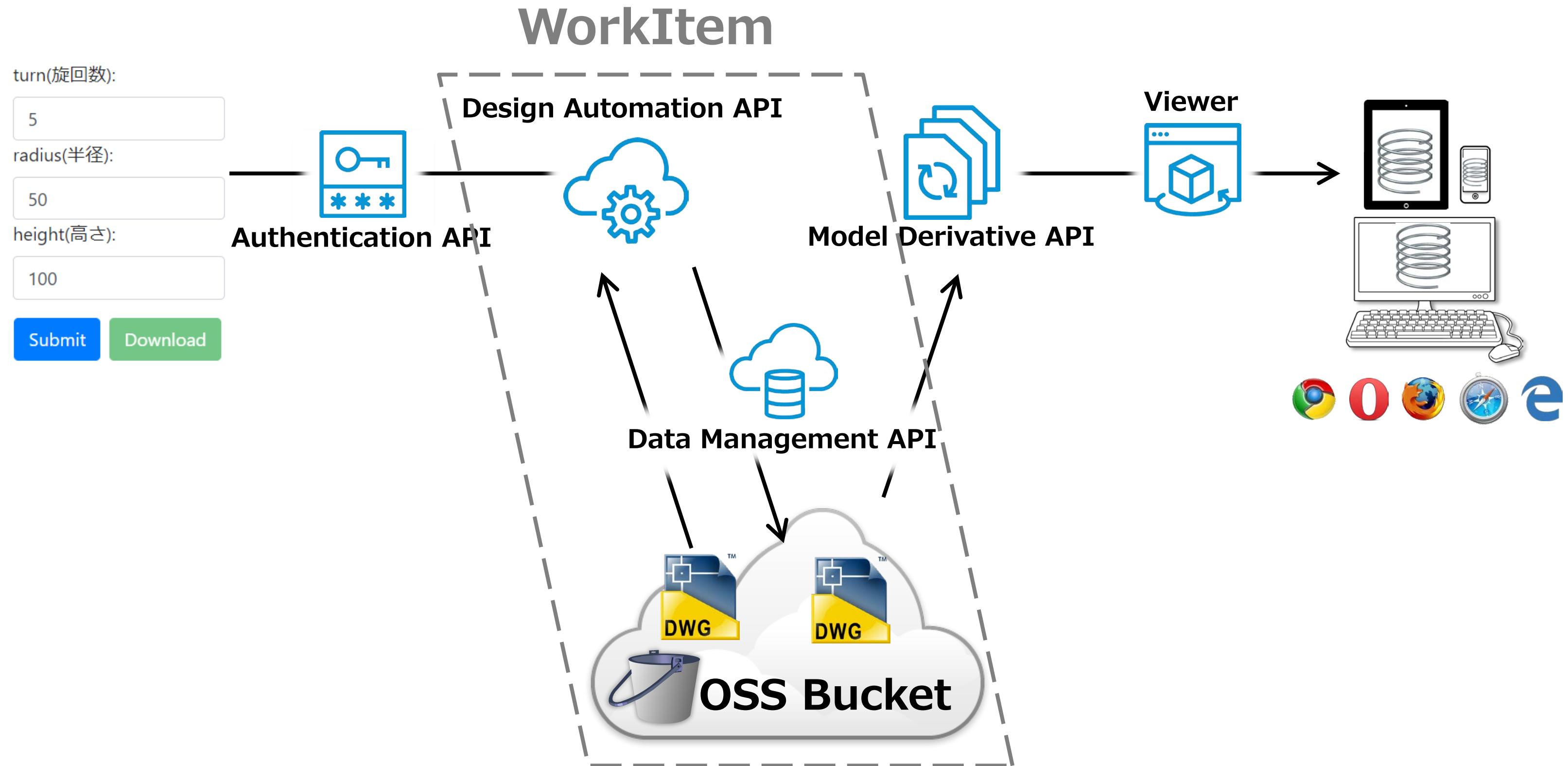
1. 'CreateCoil' AppBundle と 'dev' Alias を事前に登録
2. 'CreateCoil' Activity と 'dev' Alias を事前に登録
3. OSS Bucket に template.dwg を事前にアップロード
4. 'CreateCoil' WorkItem を登録（タスクを起動）
 - WorkItem にパラメータ（旋回数、半径、高さ）を引き渡し
 - OSS Bucket から template.dwg を作業領域にダウンロード
 - コアエンジン AcCoreConsole.exe が template.dwg をオープン
 - コアエンジンが AppBundle を作業領域に展開、アドインをロード
 - Activity の Script 記述で作業領域に result.dwg を生成、保存
 - 作業領域から result.dwg を OSS Bucket にアップロード

'CreateCoil' のワークフロー ~ つづき

5. OSS Bucket の result.dwg を SVF 変換
 - Forge Viewer へ表示
6. OSS Bucket から result.dwg をローカルをダウンロード



'CreateCoil' サンプルのワークフロー



AutoCAD アドインのオリジナル実装

- カスタム パレットからパラメータ値を取得
 - 作業フォルダから読み込み
 - アドインのあるフォルダと同じフォルダ（カレント フォルダ）

```
private void button1_Click(object sender, EventArgs e)
{
    Database acCurDb = HostApplicationServices.WorkingDatabase;

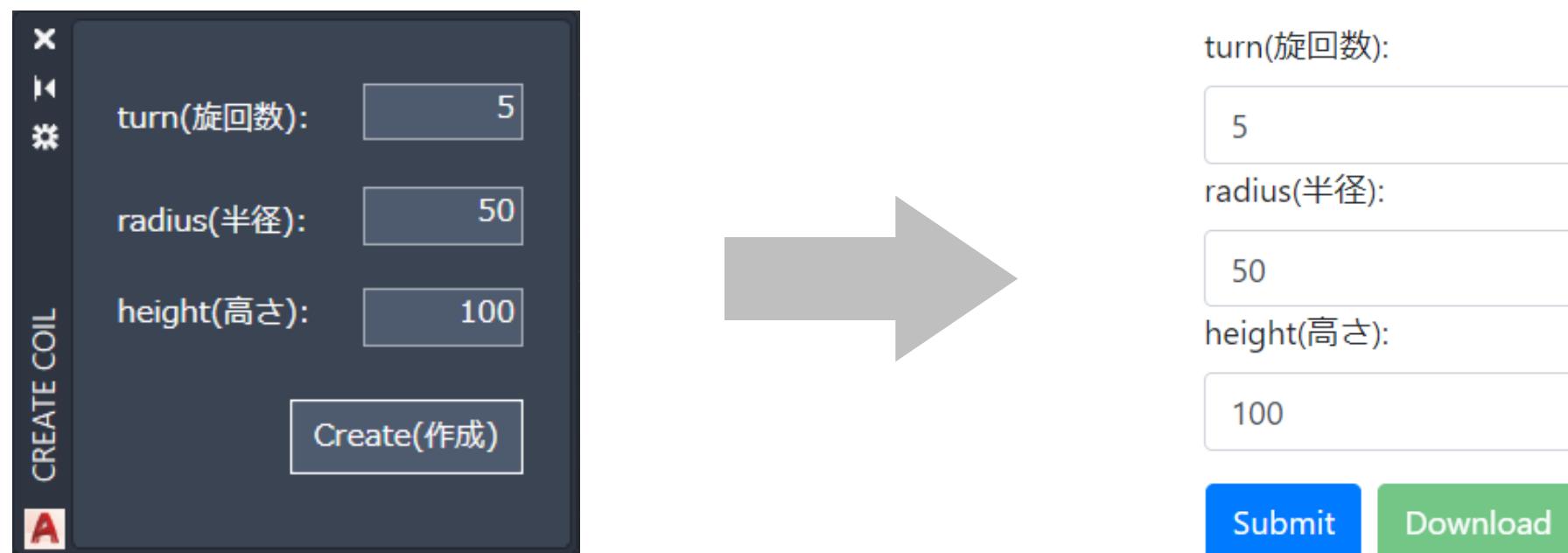
    Autodesk.AutoCAD.ApplicationServices.DocumentLock docLock =
Autodesk.AutoCAD.ApplicationServices.Application.DocumentManager.MdiActiveDocument.LockDocument();

// Get params
Int16 intTurns = short.Parse(textBox1.Text);
Double dblRadius = double.Parse(textBox2.Text);
Double dblHeight = double.Parse(textBox3.Text);

// Start a transaction
using (Transaction acTrans = acCurDb.TransactionManager.StartTransaction())
{
    :
```

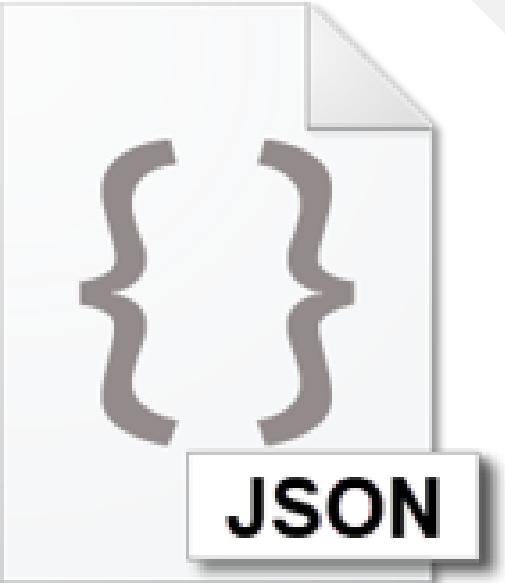
数値パラメータの受け渡し

- DA API で実行するアドインへの値の受け渡し
- カスタム コマンド実行時に参照（読み込み）させる



数値パラメータは JSON 経由で

- **JSON** (ジェイソン) : **JavaScript Object Notation**
 - 軽量で単純なデータ交換フォーマット
 - {} 括弧で囲んでデータを表現
 - JavaScript の派生ながら Web 全体でデータ交換に多用
 - Forge でも JSON を多用
 - データを Forge サーバーに渡す
 - Forge サーバーからデータを返す



Activity (アクティビティ) 登録時の JSON 例

- パラメータ値の宣言

```
{  
    "id": "CreateCoil",  
    "commandLine": [ `$(engine.path)¥¥accoreconsole.exe /i "$(args[DWGInput].path)" /al  
"$(appbundles[CreateCoil].path)" /s "$(settings[script].path)"' ],  
    "parameters": {  
        "DWGInput": {  
            "zip": false,  
            "ondemand": false,  
            "verb": "get",  
            "description": "Template drawing",  
            "required": true,  
        },  
        "Params": {  
            "zip": false,  
            "ondemand": false,  
            "verb": "get",  
            "description": "Input parameters to create coil",  
            "required": true,  
            "localName": "params.json"  
        },  
    },  
}
```

Activity (アクティビティ) 登録時の JSON 例 ~ 続き

- パラメータ値の宣言

```
"DWGOutput": {  
    "zip": false,  
    "ondemand": false,  
    "verb": "put",  
    "description": "Created drawing",  
    "required": true,  
    "localName": "result.dwg"  
},  
"settings": {  
    "script": {  
        "value": "CreateCoil\$n"  
    }  
},  
"engine": "Autodesk.AutoCAD+23_1",  
"appbundles": [ "nqpwqSDLFGkS06LgA2mvaSXy5AeH5VSJ.CreateCoil+dev" ],  
"description": "Create a coil solid to new drawing"  
}
```

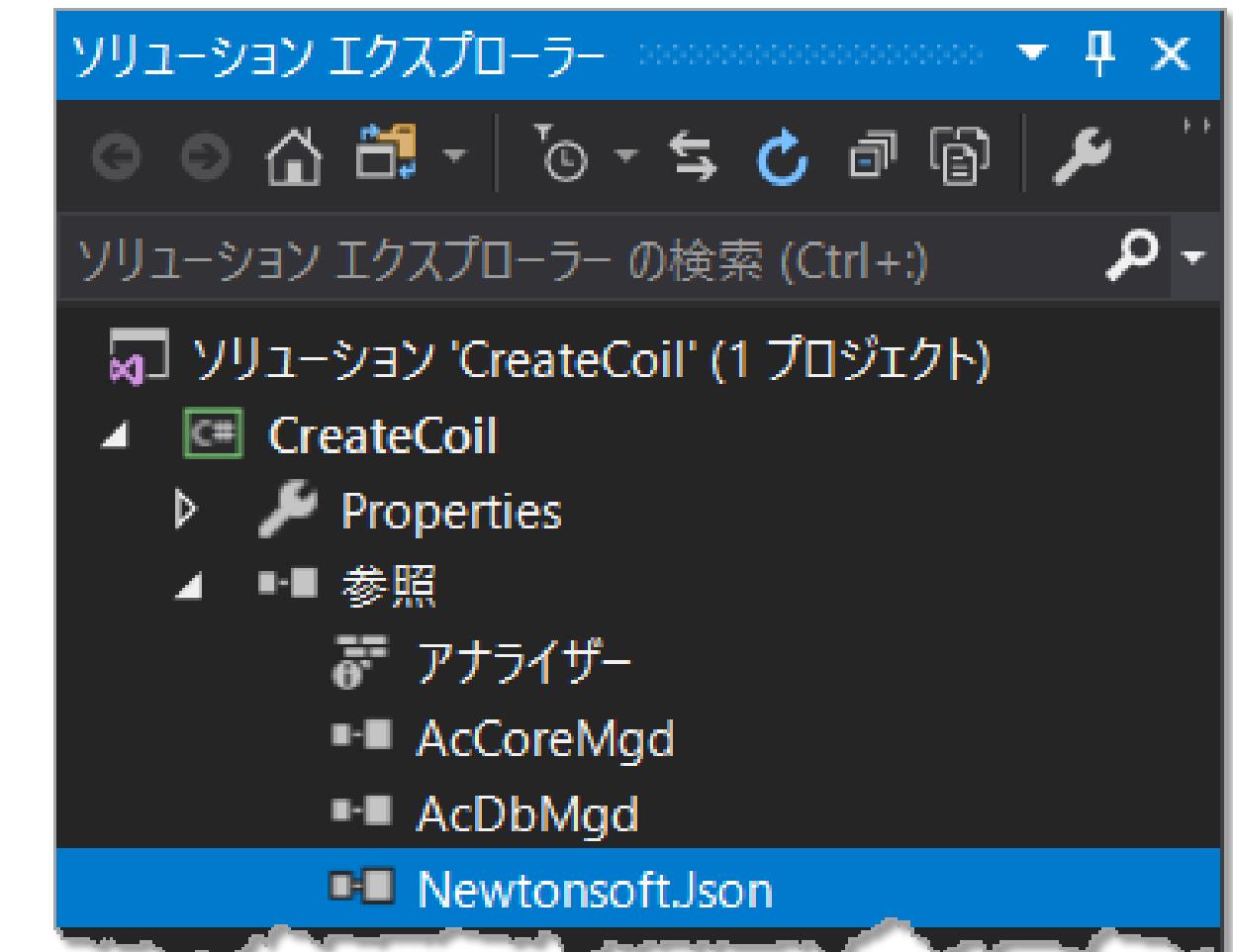
コアエンジン用アドイン？

- GUI 表示をしない実装が必須
- サポートされる API
 - ObjectARX
 - .NET API
 - AutoLISP
- .NET API の場合の参照アセンブリ
 - AcCoreMgd.dll
 - AcDbMgd.dll
- ...
- Newtonsoft.json

推奨

必須
AcMgd.dll 参照不可

ご参考：NuGet から入手可能
(JSON ファイル操作で利用)



アドイン (AppBundle) の改造・移植

- 作業フォルダの JSON ファイルからパラメータ値を読み込み

```
[CommandMethod("CreateCoil", CommandFlags.Modal)]
public void MyCommand() // This method can have any name
{
    Log("¥nStart Addin process . . .");

    // Put your command code here
    Database acCurDb = HostApplicationServices.WorkingDatabase;

    Log("¥nGot database . . .");

    // Get params from JSON file
    InputParams inputParams =
        JsonConvert.DeserializeObject<InputParams>(File.ReadAllText(".¥¥params.json"));
    Double intTurns = inputParams.Turn;
    Double dblRadius = inputParams.Radius;
    Double dblHeight = inputParams.Height;
    Log("¥nAddin retrieves Turn:{0}, Radius{1}, Height:{2}", intTurns, dblRadius, dblHeight);
    :
}
```

自動ローダーの PackageContents.xml

CreateCoil.bundle



PackageContents.xml



Contents



CreateCoil.dll



Newtonsoft.Json.dll



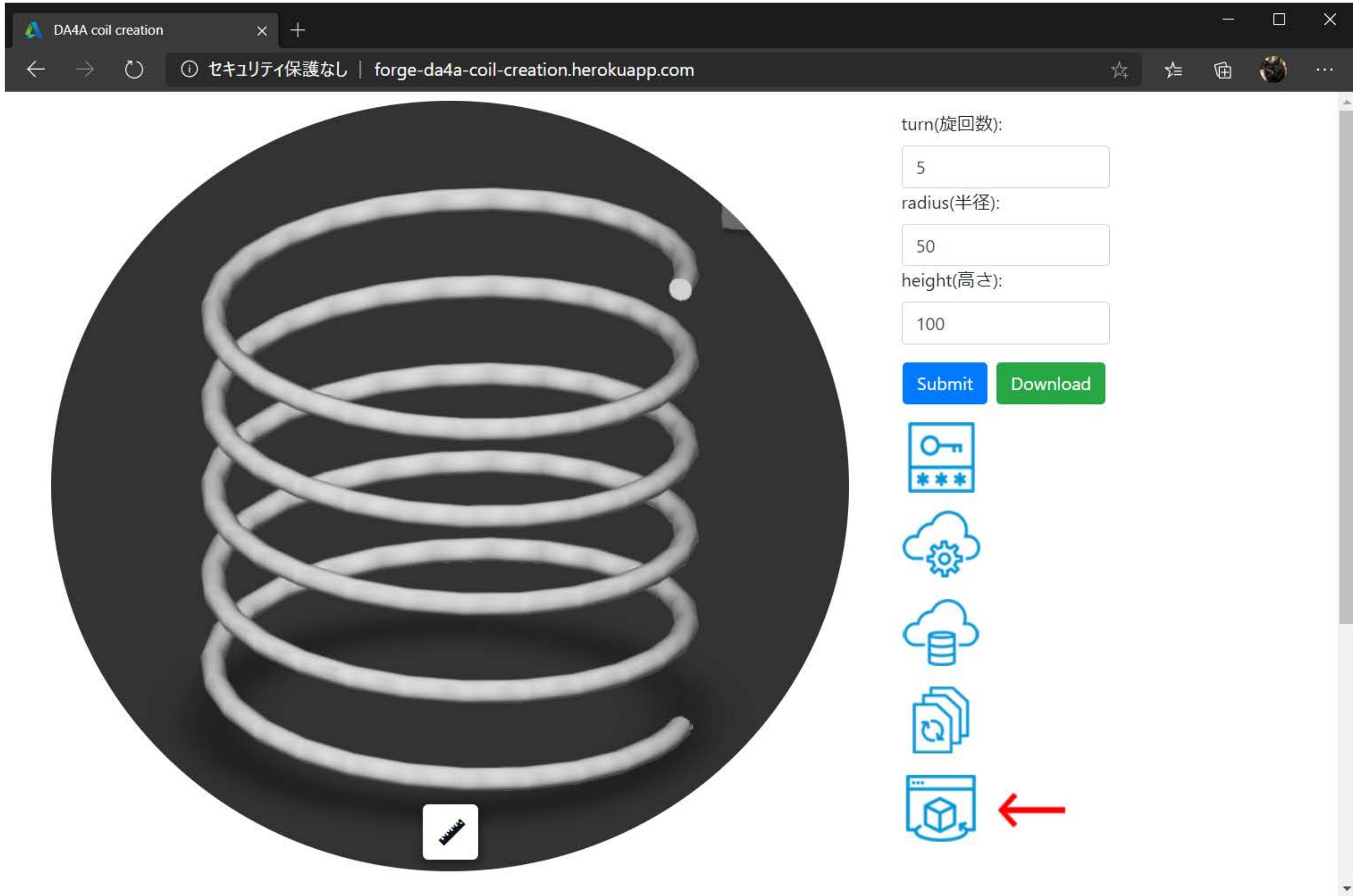
Newtonsoft.Json.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ApplicationPackage
    SchemaVersion="1.0"
    AppVersion="1.0"
    ProductCode=" "
    Name="CreateCoil Package"
    ProductType="Application"
    Description="Create a Coil"
    Author="Toshiaki Isezaki" >
    <CompanyDetails
        Name="Autodesk Ltd., Japan"
        Url="https://www.autodesk.co.jp"
        Email="xxxxxxxx.xxxxxxx@autodesk.com" />
    <Components>
        <RuntimeRequirements
            OS="Win64"
            Platform="AutoCAD"
            SeriesMin="R23.0"
            SeriesMax="R24.0" />
        <ComponentEntry
            AppName="CreateCoil"
            ModuleName="../Contents/CreateCoil.dll"
            AppDescription="Create a Coil"
            LoadOnAutoCADStartup="True" />
    </Components>
</ApplicationPackage>
```

AppBundle のアップロード

1. POST **appbundles** endpoint でアップロード情報を取得
 - アップロード先 URL、アクセスに必要な Policy、Credentials
2. FormData としてパッケージバンドルをアップロード
 - アドイン本体 + PackageContents.xml を ZIP 圧縮

```
var uri = JSON.parse(JSON.stringify(uploadParameters)).endpointURL;
let formData = new FormData();
formData.append('key', fdata['key']);
formData.append('content-type', fdata['content-type']);
formData.append('policy', fdata['policy']);
formData.append('success_action_status', fdata['success_action_status']);
formData.append('success_action_redirect', fdata['success_action_redirect']);
formData.append('x-amz-signature', fdata['x-amz-signature']);
formData.append('x-amz-credential', fdata['x-amz-credential']);
formData.append('x-amz-algorithm', fdata['x-amz-algorithm']);
formData.append('x-amz-date', fdata['x-amz-date']);
formData.append('x-amz-server-side-encryption', fdata['x-amz-server-side-encryption']);
formData.append('x-amz-security-token', fdata['x-amz-security-token']);
formData.append("file", fs.createReadStream(req.file.path));
formData.submit(uri, function (error, uploadres) {
```



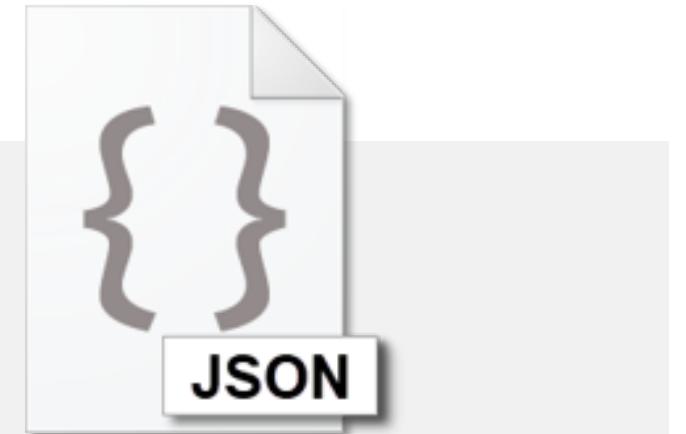
WorkItem (ワークアイテム) 実行時の JSON

```
{  
  "activityId": "nqpwqsDLFGkS06LgA2mvaSXy5AeH5VSJ.CreateCoil+dev",  
  "arguments": {  
    "DWGInput": {  
      "url": "https://developer.api.autodesk.com/oss/v2/signedresources/xxxxx?region=US",  
      "headers": {  
        "Authorization": "Bearer xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",  
        "Content-type": "application/octet-stream"  
      },  
      "verb": "get"  
    }  
    "Params": {  
      "url": "data:application/json,{\"turn\":\"17\", \"radius\":\"70\", \"height\":\"250\"}"  
    },  
    "DWGOutput": {  
      "url": "https://developer.api.autodesk.com/oss/v2/signedresources/xxxxx?region=US",  
      "headers": {  
        "Authorization": "Bearer xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",  
        "Content-type": "application/octet-stream"  
      },  
      "verb": "put"  
    }  
  }  
}
```

WorkItem 実行時に起こること

- AppBundle (アドイン) は実行時に作成の作業フォルダに展開
- DA API は WorkItem で指定された値を利用
 - 指定のクラウドストレージから DWG をダウンロード (任意)
 - 指定の値を持つ JSON ファイルを作成

```
{  
    "turn" : "17",  
    "radius" : "70",  
    "height" : "250"  
}
```



- 成果ファイルを指定のクラウドストレージへ保存 (アップロード)
 - DWG や PDF など
- WorkItem 終了後には AppBundle、ファイルは削除されます
 - キャッシュして後日の WorkItem で使用することは不可

ちなみに コアエンジンに DWG をオープンさせるには

- accoreconsole.exe の起動オプションを利用

```
C:\Program Files\Autodesk\AutoCAD 2020\accoreconsole.exe
Version Number: Q.47.0.0 (UNICODE)
Usage:
AcCoreConsole.exe [/i <input_dwg>] /s <script>[/product <product>] [/l <language>] [/isolate <userId> <userDataFolder>]
[/readonly] [/profile] <profile>
```

- Activity (アクティビティ) 登録時の JSON 抜粋

```
{
  "id": "CreateCoil",
  "commandLine": ["$(engine.path)\accoreconsole.exe /i "$(args[DWGInput].path)" /al
"$(appbundles[CreateCoil].path)" /s "$(settings[script].path)"'],
  "parameters": {
    "DWGInput": {
      :
    }
  }
}
```



ちなみに DWG を保存させるには

- アドイン側で保存処理を実装

```

    :
    // Save the new object to the database
    acTrans.Commit();
}

acCurDb.SaveAs(".\result.dwg", DwgVersion.Current);
Log("result.dwg was saved ...");
}

```

- Activity (アクティビティ) 登録時の JSON 抜粋

```

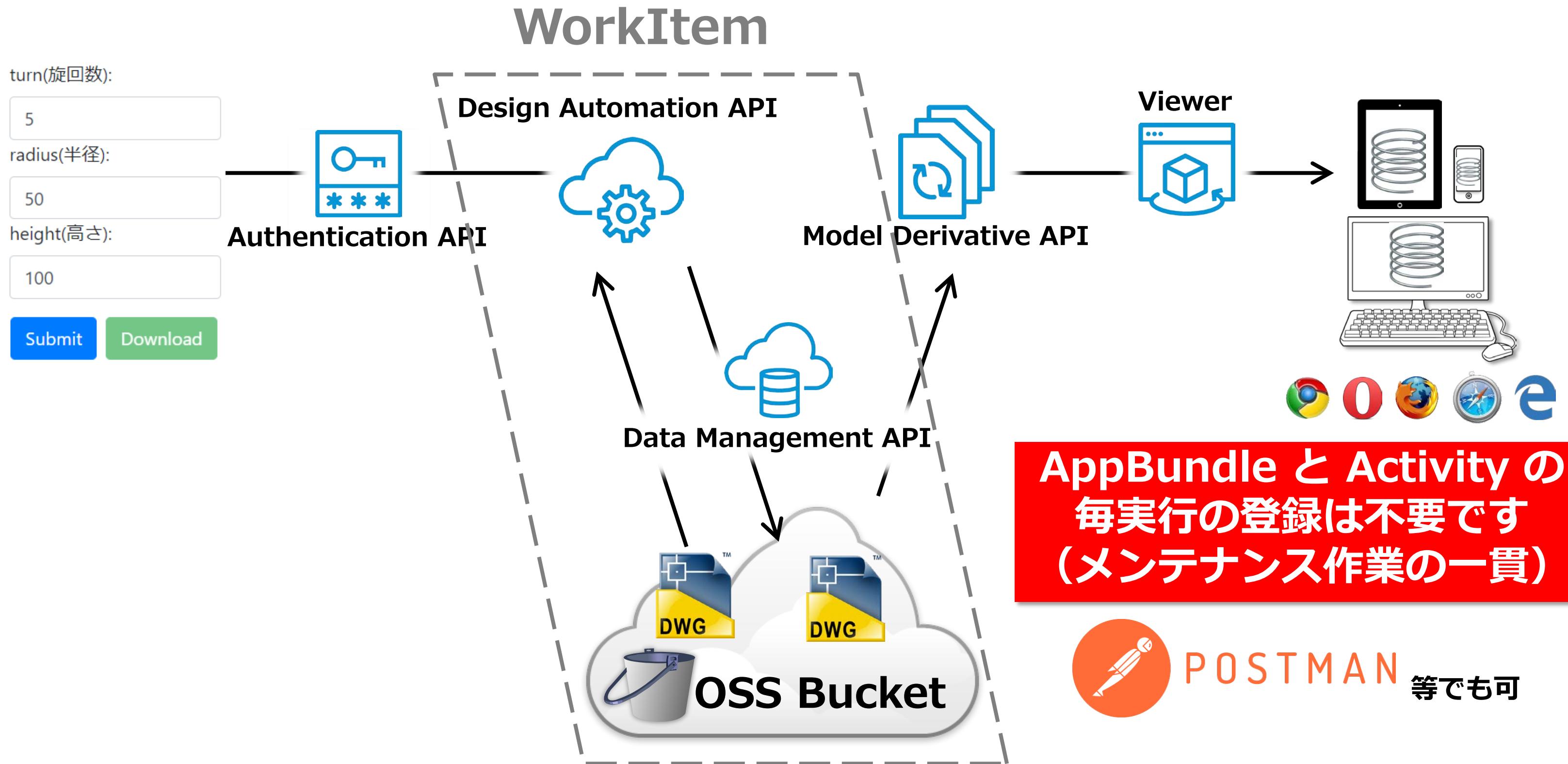
"DWGOutput": {
    "zip": false,
    "ondemand": false,
    "verb": "put",
    "description": "Created drawing",
    "required": true,
    "localName": "result.dwg"
}

```

:



'Create Coil' サンプル実行時のワークフロー





AUTODESK®

Make anything.