Table of Contents

[Overview 2](#_Toc101521156)

[Implementation 2](#_Toc101521157)

[Preparation 2](#_Toc101521158)

[**CmdGroundTruth** 2](#_Toc101521159)

[**CmdValidation** 2](#_Toc101521160)

[Storage 3](#_Toc101521161)

[Extensible Storage 3](#_Toc101521162)

[Validation 4](#_Toc101521163)

[Manually 4](#_Toc101521164)

[Automatically 4](#_Toc101521165)

[Loophole 4](#_Toc101521166)

[User Interface 5](#_Toc101521167)

[Improvements 6](#_Toc101521168)

# Overview

What is RvtLock3r?

RvtLock3r is a Revit .NET C# custom add-in designed to keep track of any modifications or alteration done on a BIM element.

It is purposely meant to validate that certain BIM element properties have not been modified.

Revit does not provide any functionality to ensure that model properties / parameter values are not modified.

# Implementation

## Preparation

The add-in implements two commands: CmdGroundTruth and CmdValidation.

### **CmdGroundTruth**

Initialize the ground truth data for a given model.

The command is fully manual and designed for the vendor.

The ground truth data contains a list of triples:

* ElementId
* shared parameter GUID
* Checksum

The vendor can manipulate the command and have the power to decide what data they want to store in the ground truth.

### **CmdValidation**

Performs the validation against the stored ground truth data stored above.

The command reads a set of ground truth data from some storage.

The add-in iterates over all elements and parameters specified by these triples.

Reads the corresponding parameter value.

Calculates its checksum and validates it by comparing with the ground truth value.

Discrepancies are logged and a report is presented to the user.

The add-in does not care what kind of elements or parameters are being examined. That worry is left up to whoever creates the ground truth file.

## Storage

### Extensible Storage

The ground truth data is stored in the Extensible storage using Named Ground Truth Storage.

References for the using Named Ground Truth Storage is [Named Guid Storage for Project Identification](https://thebuildingcoder.typepad.com/blog/2016/04/named-guid-storage-for-project-identification.html).

Basically, it allows to simply create own Ground Truth for the current Revit project and use the same to identify it globally forever after.

I define an extensible storage schema named **NamedGroundTruthStorage** that just stores one single Ground Truth object.

To create a new project identifier, I create a Ground Truth and store it in an extensible storage entity with the above schema on a Revit DataStorage element with a specific element name, e.g., ground\_truth\_identifier\_v0.

To search for an existing project identifier, I can filter for all data storage elements with extensible storage entities containing data matching our specific schema and with the given element name.

## Validation

Two approaches have been provided to accomplish the validation of the ground truth.

### Manually

Using CmdValidation: The vendor can initiate this command manually by clicking the push button Validate under the ribbon tab Lock3r.

### Automatically

Using Event drive approach.

#### DocumentOpened

Using App application, OnDocumentOpened event is invoked. The event gets the active document right after its opened, runs the validation by cross reading the ground truth from extensible storage against the elements of the active document.

If a discrepancy is encountered, the application presents an informational message to the end-user that the document is corrupted and contains tampered information.

It is the choice of the user to react on the message whether to proceed working with the corrupted model or contact the vendor for an original file.

#### DocumentSaving

The application invokes, OnDocumentSaving event every time an end-user attempts to save a modified model during interaction whether intended or not.

A warning is presented to the end user that he /she is prohibited to modify a certain property and saving is cancelled.

If user decides to terminate / close Revit after a modification, OnDocumentSaving is again invoked providing user with option whether to save the changes or not.

If the user decides not to save the changes and clicks ‘No’ the document does not persist the modified elements. The originality is maintained.

### Loophole

If the user decides to save and click ‘Yes’ the model is modified!

## User Interface

The add-in provides a Ribbon Tab on Revit User Interface called Lock3r.

The ribbon panel contains two push buttons:

1. Ground Truth Button:

Generates the ground Truth data.

1. Validate Button:

Implements the validation method manually to check whether any modification was done.

# Improvements

* Implement [DMU dynamic model updater](https://thebuildingcoder.typepad.com/blog/about-the-author.html#5.31) to prevent modification of the protected parameter values
* Implement [end user settings](https://thebuildingcoder.typepad.com/blog/2016/09/hololens-escape-path-waypoint-json-exporter.html) to choose validation strategy: command / opening and closing events / DMU
* Migrate to Forge Design Automation for Revit