Legacy Data Analysis on Web with Forge – Cost Prediction in Initial Design Stage

TAKAHIRO Oka

NIHONSEKKEI, INC.

Class Summary

The class presents cost management strategy using web based system that includes Forge.

In the system, we focused to analyze legacy project data interactively for on-going project with Machine Learning Algorithm In an attempt of 5D BIM. It supports decision-making of cost control at the initial design stage when Revit model don't have detail object for cost estimation. By improving quality of cost estimate in initial design stage, it is easy to manage the cost within target budget through project.

We will focus on as billow;

- How to get past project data interactively through WebAPI of internal system.
- How to get and share on-going project data with "Data management API " and "Viewer" of Forge.
- Data analysis techniques for cost estimation of on-going project using "scikit-learn" that is machine learning library in Python.
- Data cycle system to improve quality of analysis, constantly.

Key Learning Objectives

- Describe effective cost management workflow at initial design stage for BIM project.
- Strategy of Utilization legacy project data with WebAPI of internal systems.
- Picking up the data of on-going project on Web using Forge.
- Data Analysis method for decision-making with “Scikit-learn” that is library of Python.
- About framework and library to make forge application.

About NINHONSEKKEI, INC.

Establishment – September 1, 1967

Employee – 951(May 2018)

Scope of Services

1. Planning, Design, Supervision and Survey for projects related to Architecture, Landscape, Civil Engineering, Structural Engineering and Interiors.
2. City and Regional Planning, Urban Redevelopment Planning and Environmental Assessment Surveys.
3. Surveys and Planning for Information Technology and Management.
4. All services related or incidental to the above.

Achievement

1960’s

1698 Kasumigaseki Building

1970’s

1674 Shinjyuku Mitsui Building

1980’s

1987 Tama Zoo Insectarium

1990’s

1995 ACROS Hukuoka

2000’s

2000 Ajinomoto Stadium

2000 Aquamarin Hukushima

2005 Nihonbashi Mitsui Tower

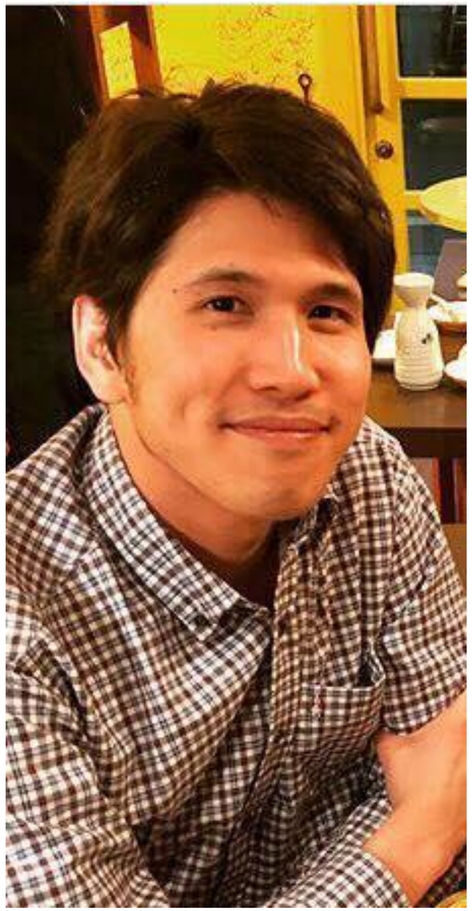
2002 Yamaguchi Kirara Dome

2006 National Art Center,Tokyo

2010～

2014
Toranomon Hills

2017
Akasaka Inter city Air



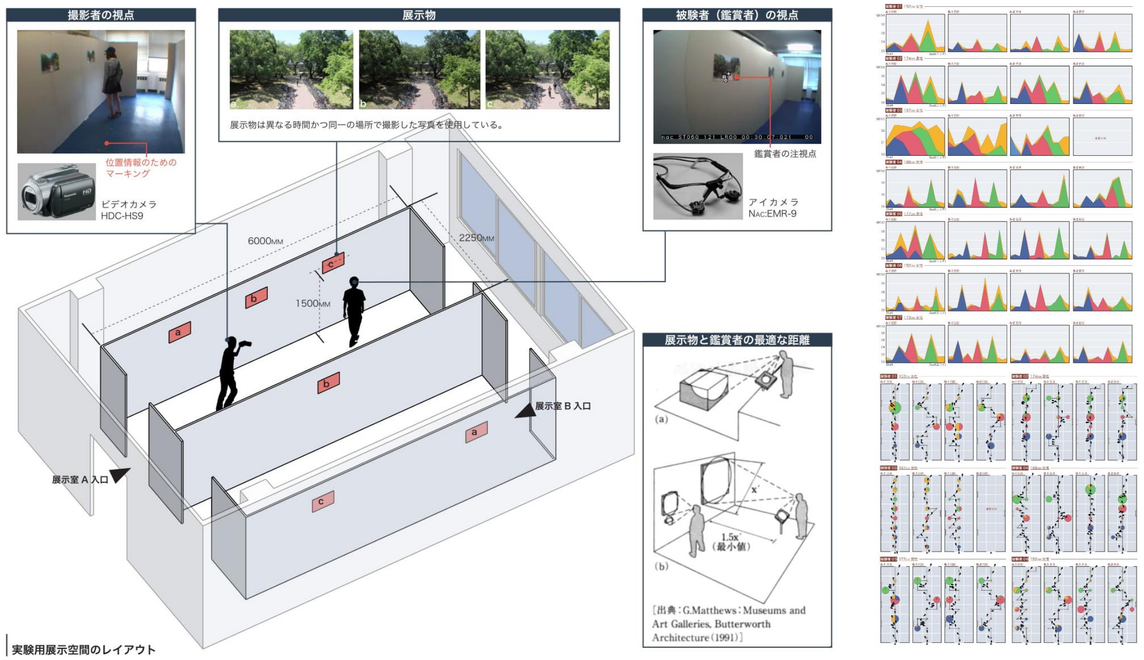
About the speaker

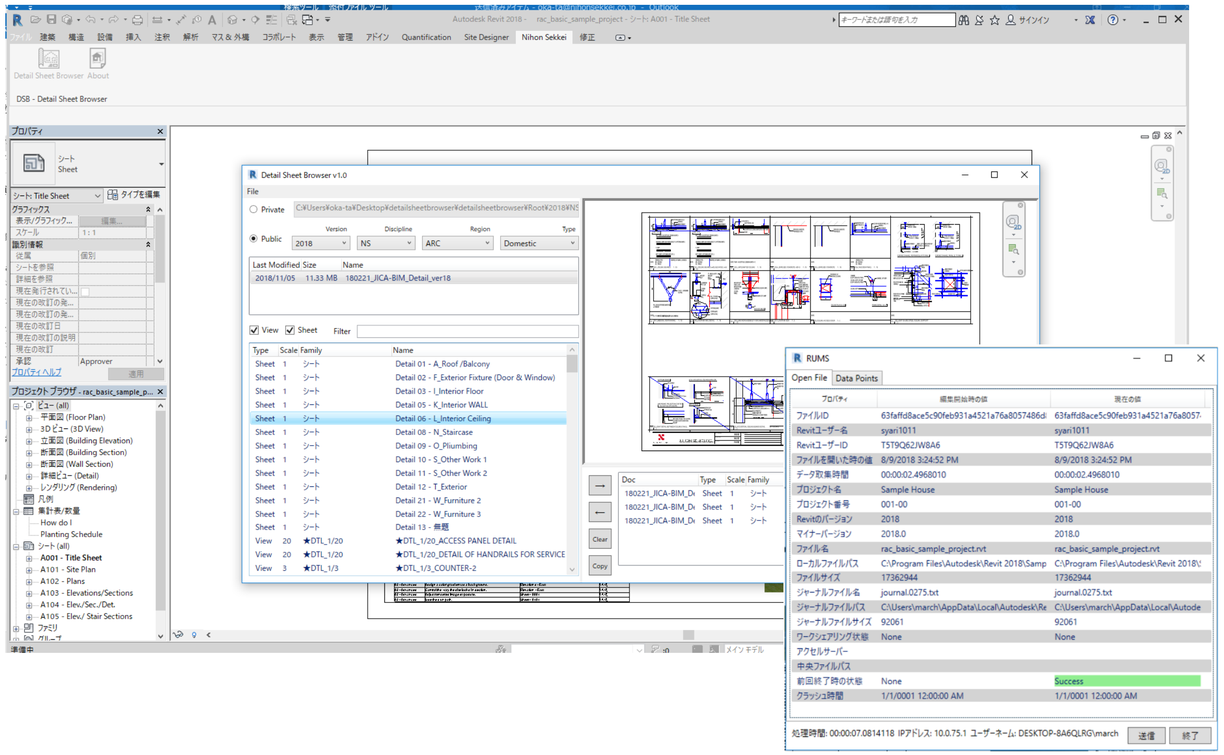
TAKAHIRO Oka

I’m Software Engineer for the NIHON SEKKEI, Inc, where I had developed plug-in with Revit API for Revit, and designed Web API for internal system corporation, analyzes data.

I studied spatial perception for the exhibition space using the eye tracking and VR device at the Chiba University.

I began using Revit and Dynamo from 2016 after graduation. Currently, I have developed various tools focusing on how to use legacy project data for on-going projects with web technology and data analysis in NIHON SEKKEI, Inc





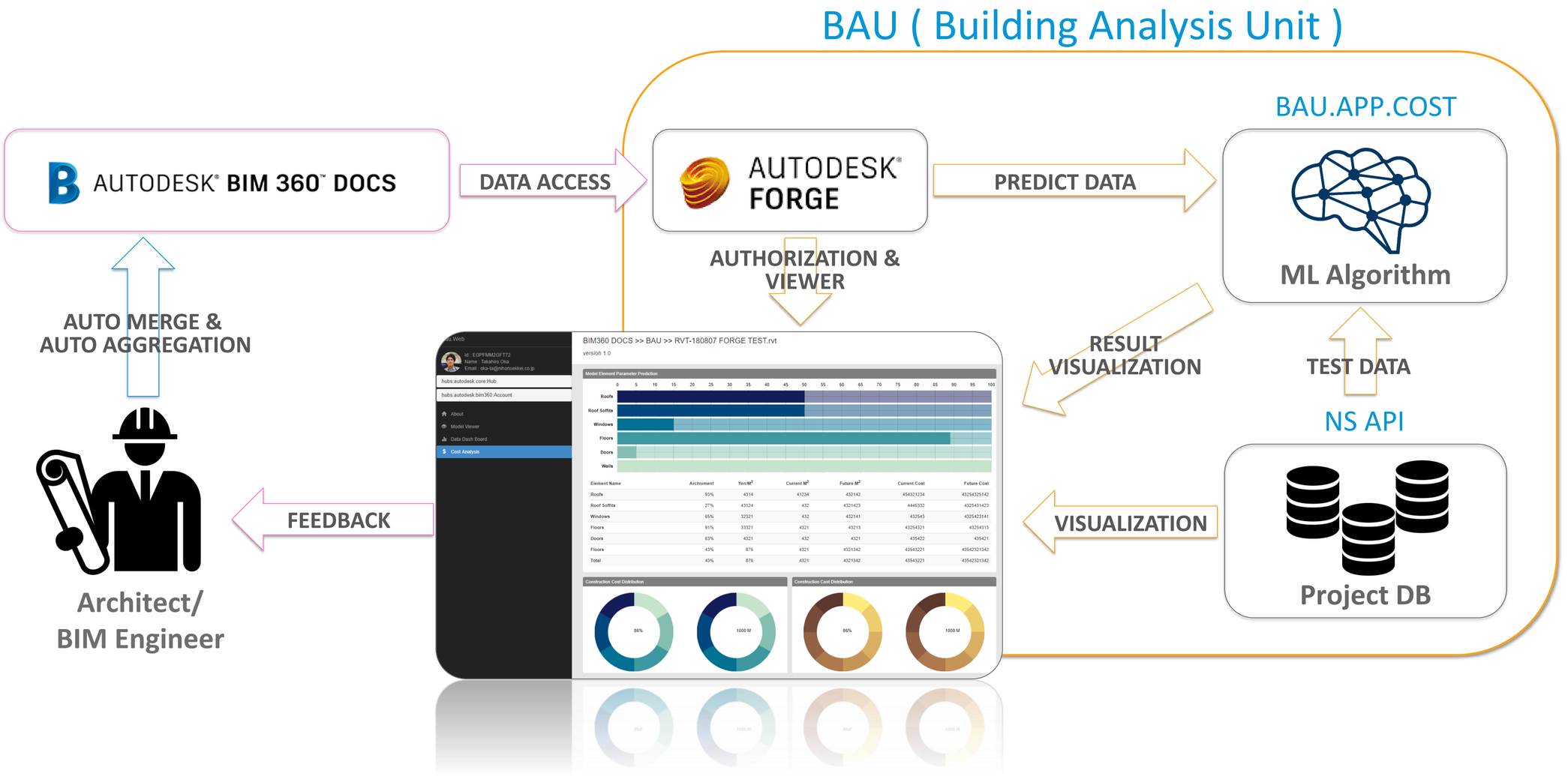
Study about Spatial Perception

Study about how behave human in exhibition space. In this study, I use eye-tracking device and VR device to collect some data such as sight point , subject’s neck angle, subject position, etc. and analyzed them to find pattern of behavior.

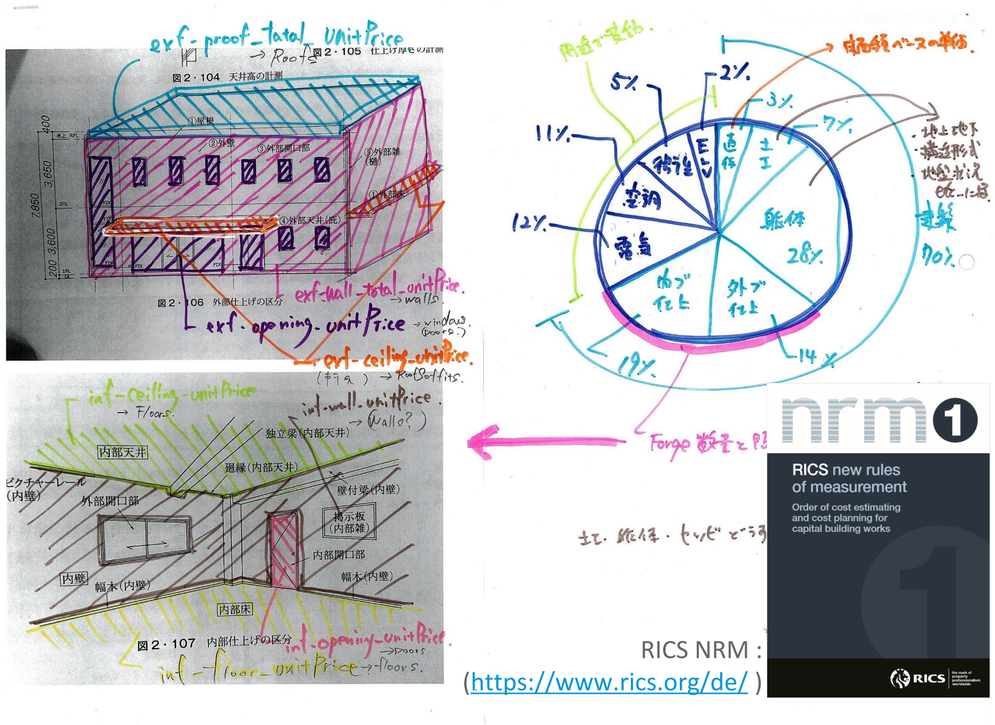
BIM Tool Development

Some application to support drawing and modeling in Revit. And, Revit Monitoring System, which collect Revit data when user open them, to prevent some errors by analysis error pattern.

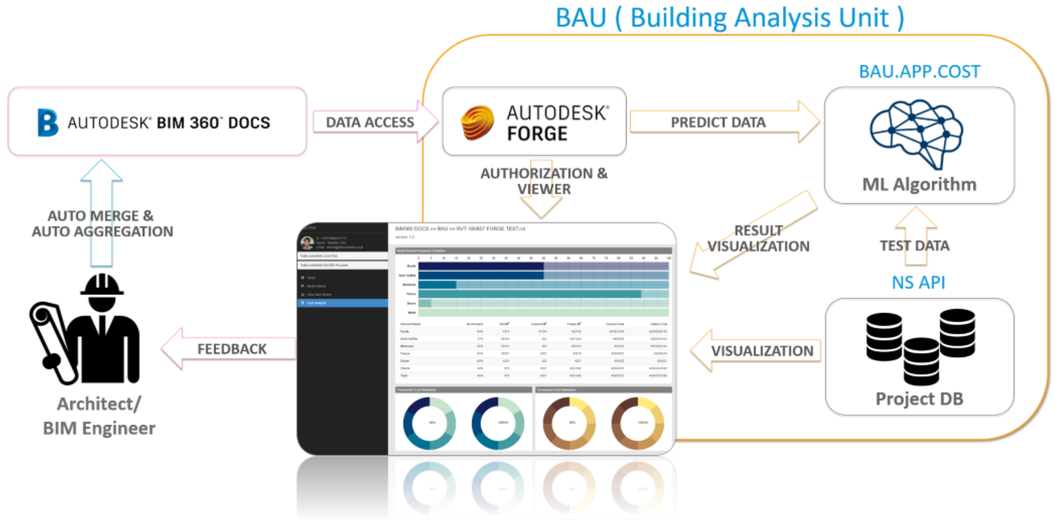
System Overview



Construction Cost Estimation with Machine Learning instead of Rule Based System



RICS NRM :
(https://www.rics.org/de/)



RULE BASED ESTIMATION

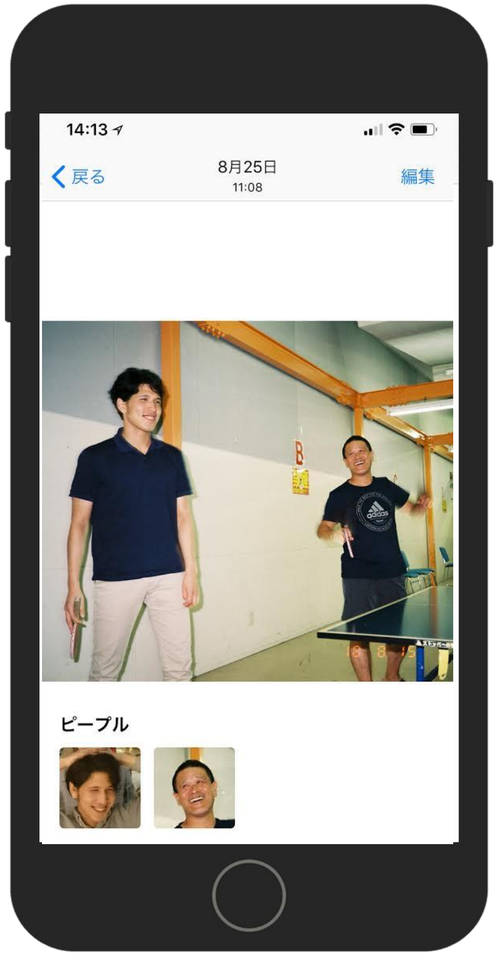
Construction Cost is estimated by rule based on Knowledge and Experience of people.

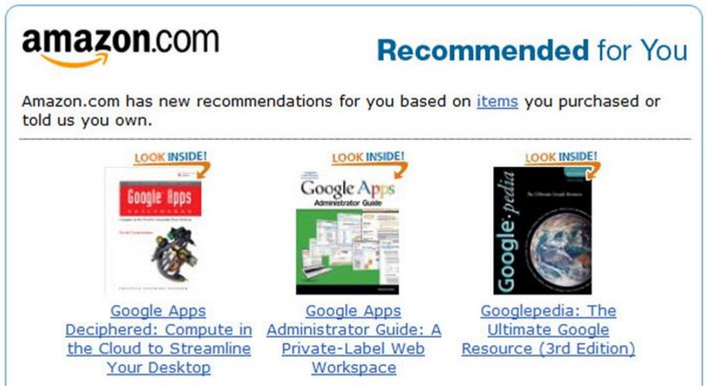
MACHINE LEARNING ESTIMATION

In Machine Learning concept, rule is found by analyzing data with algorithm.

Demand of Big Data

Recently, growth of Machine Learning technology and API Economy accelerate increasing demand of Big Data, and based on the data, many services and researches are publishing at various department. As example, picture that take people were used for Machine Learning to generate analysis model of detecting people face, and the model have been incorporating into camera of smart phone device. And some item that you unintended but you want are recommended when You buy something at web shopping site.



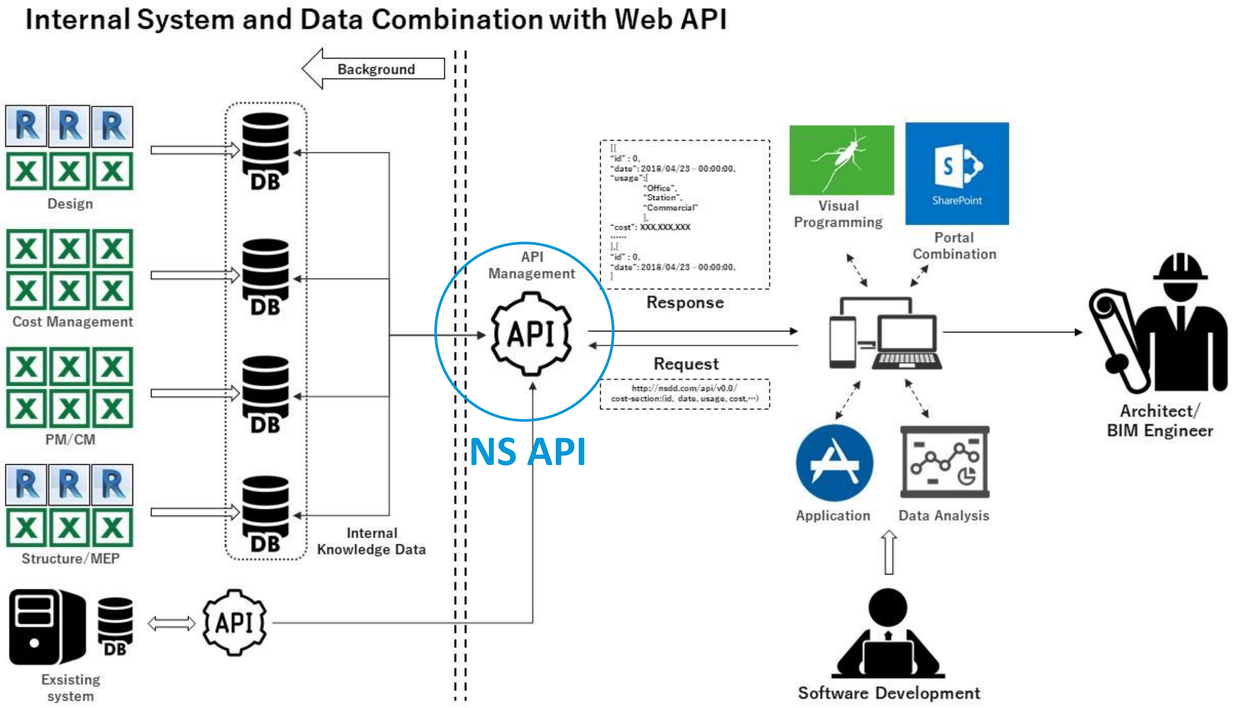


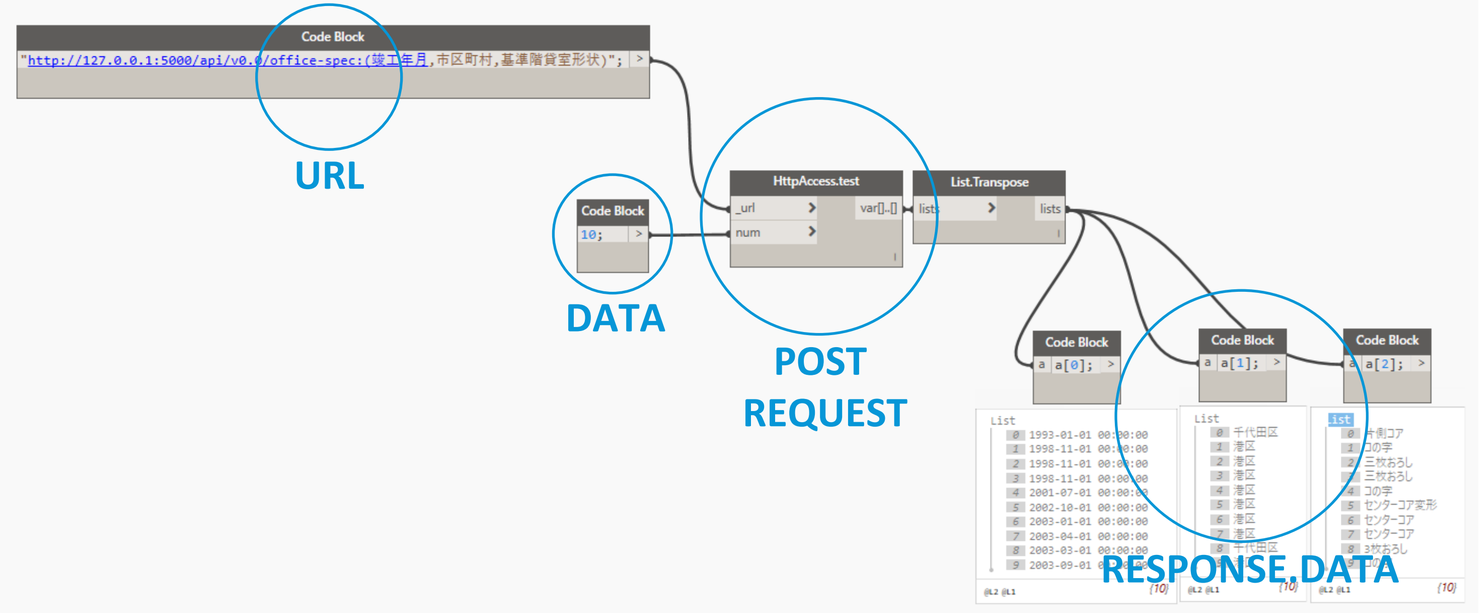
: Amazon
(Source : https://www.amazon.co.jp/)

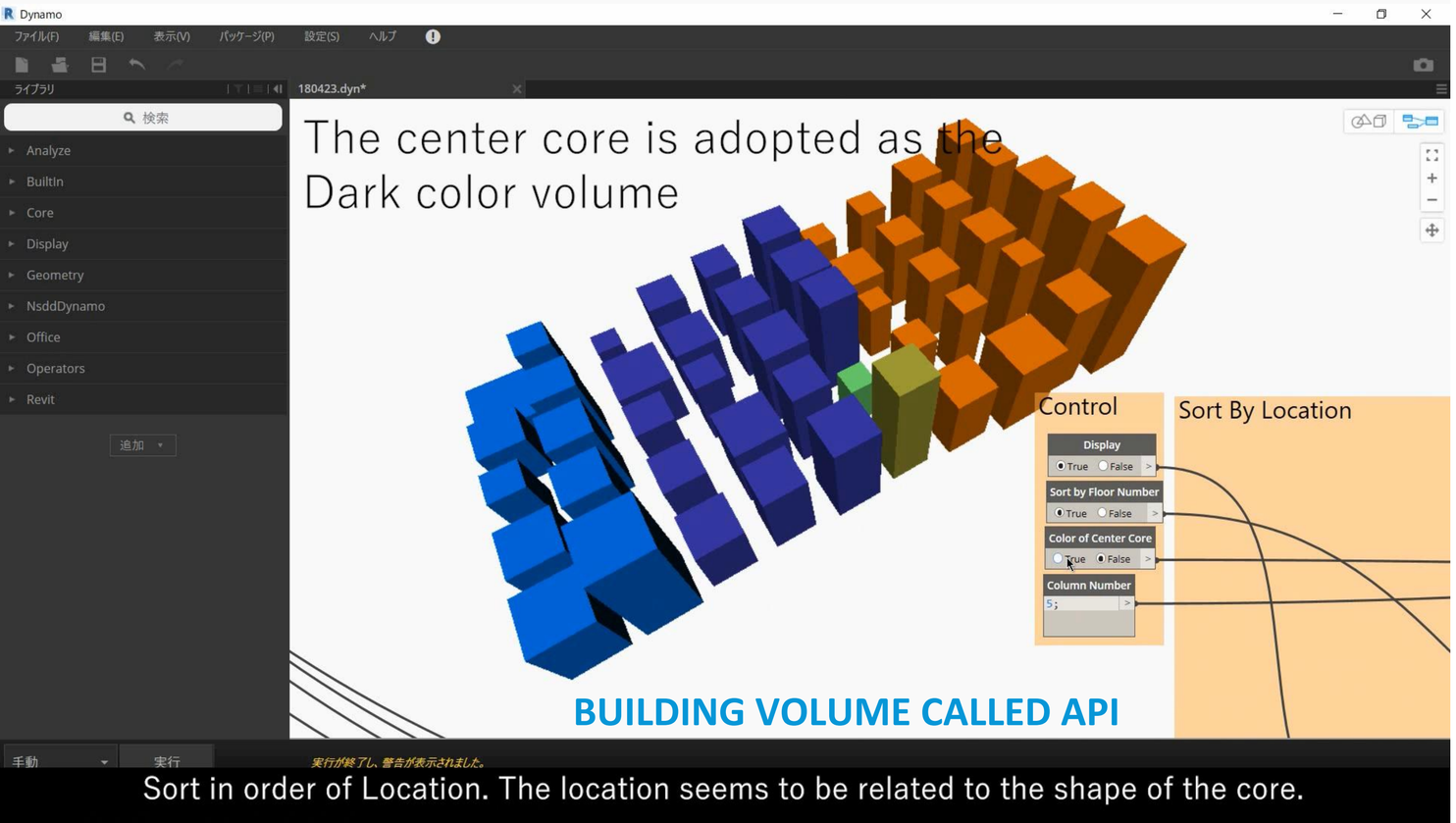
: Google Home Speaker
(Source : https://store.google.com/jp/product/google_home)

What is Big Data for us? How use it ?

NIHONSEKKEI have been building a system to access accumulated project data from various platform easily, which called **NS API**. We can access accumulated project data with HTTP protocol by NS API which is Web API like Forge API, and this system accelerate collaboration with each internal systems and application development , data analysis, etc.





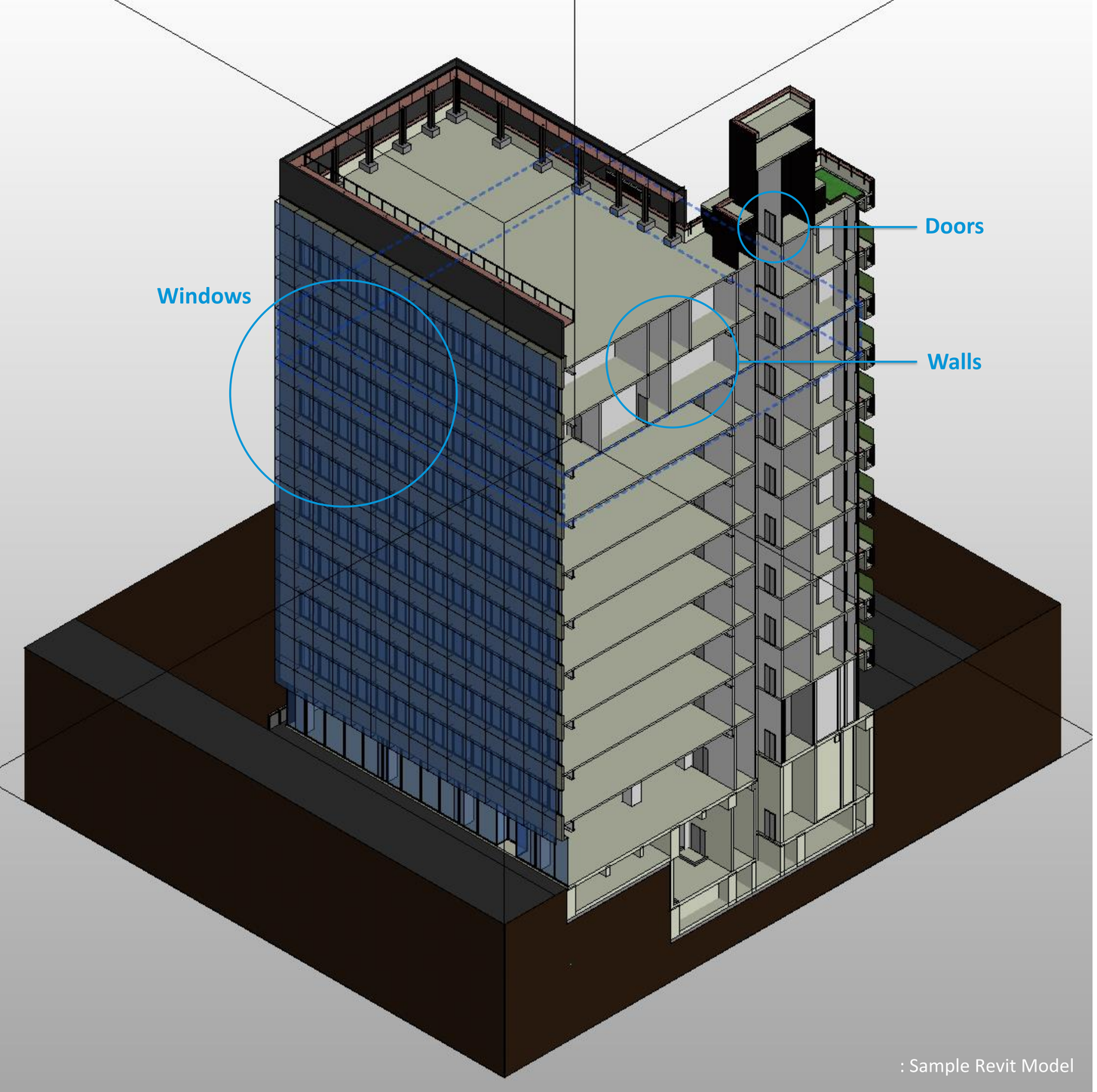


Sort in order of Location. The location seems to be related to the shape of the core.

Dynamo × NS API

One way to use NS API, we made dynamo custom node for using HTTP Protocol in Dynamo. NS API approve to access past project database from Dynamo so that We can utilize the data ,which is such as curtain panel size and building volume , to review physical design for on-going Revit project.

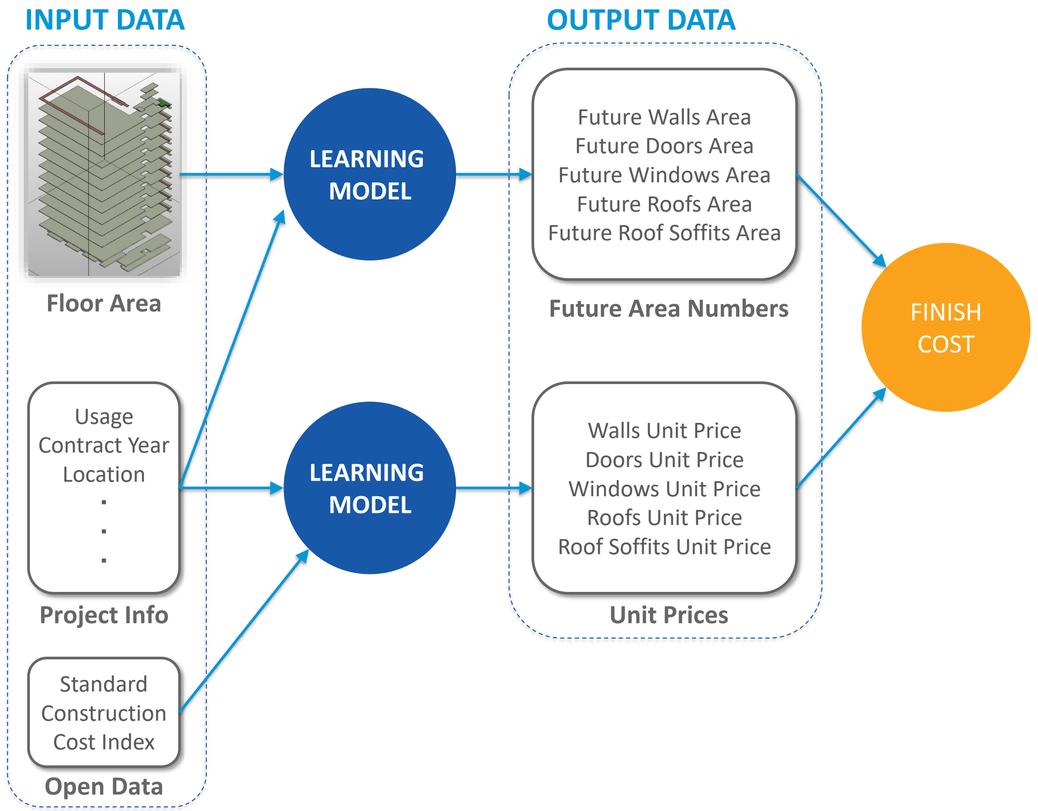
Process of Construction Cost Estimation with ML

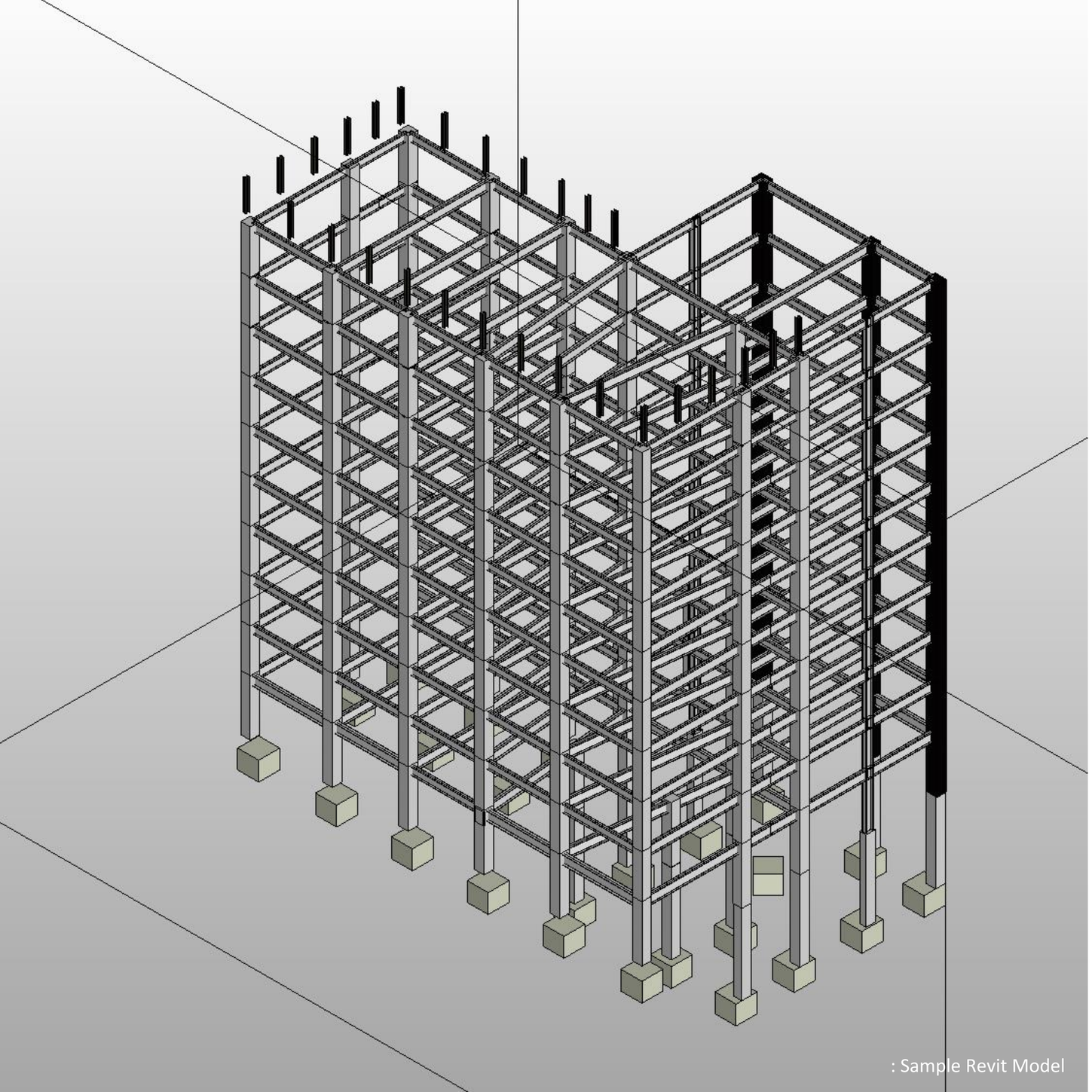


: Sample Revit Model

Cost Estimation - 1

Finish Cost Estimation

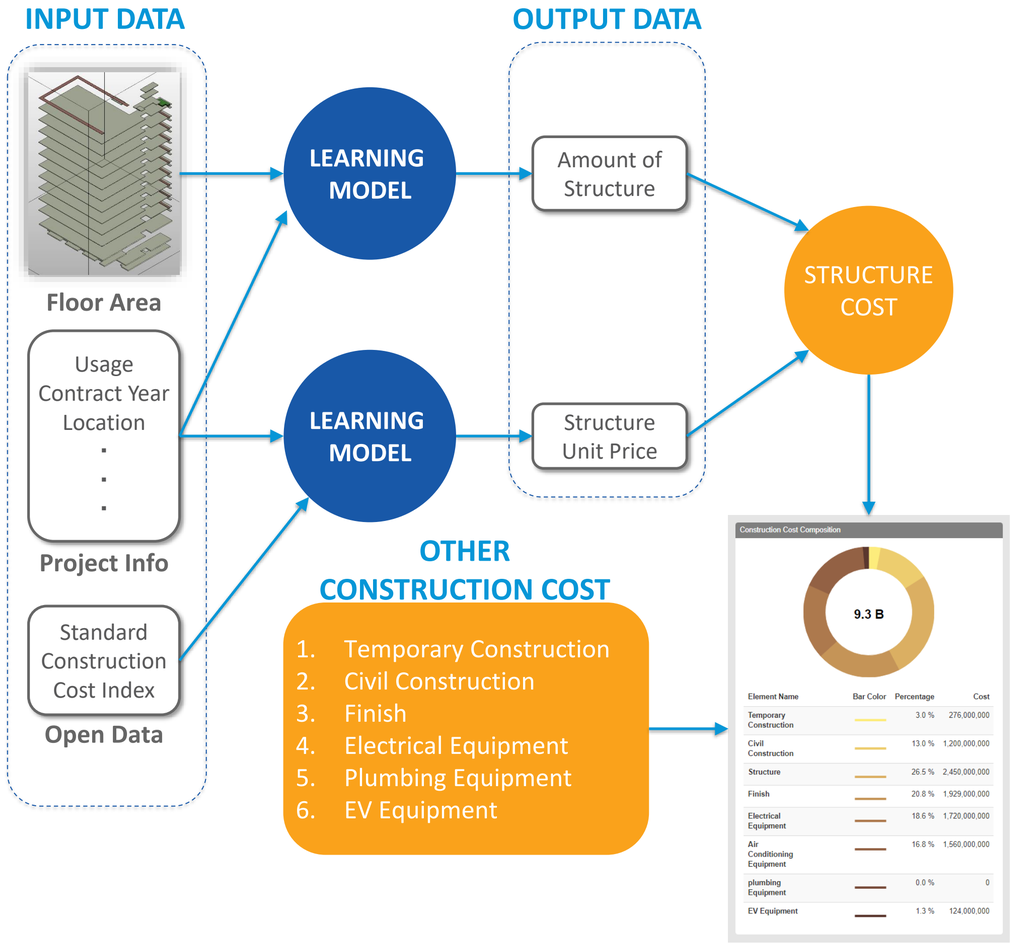




: Sample Revit Model

Cost Estimation - 2

Construction Cost Estimation



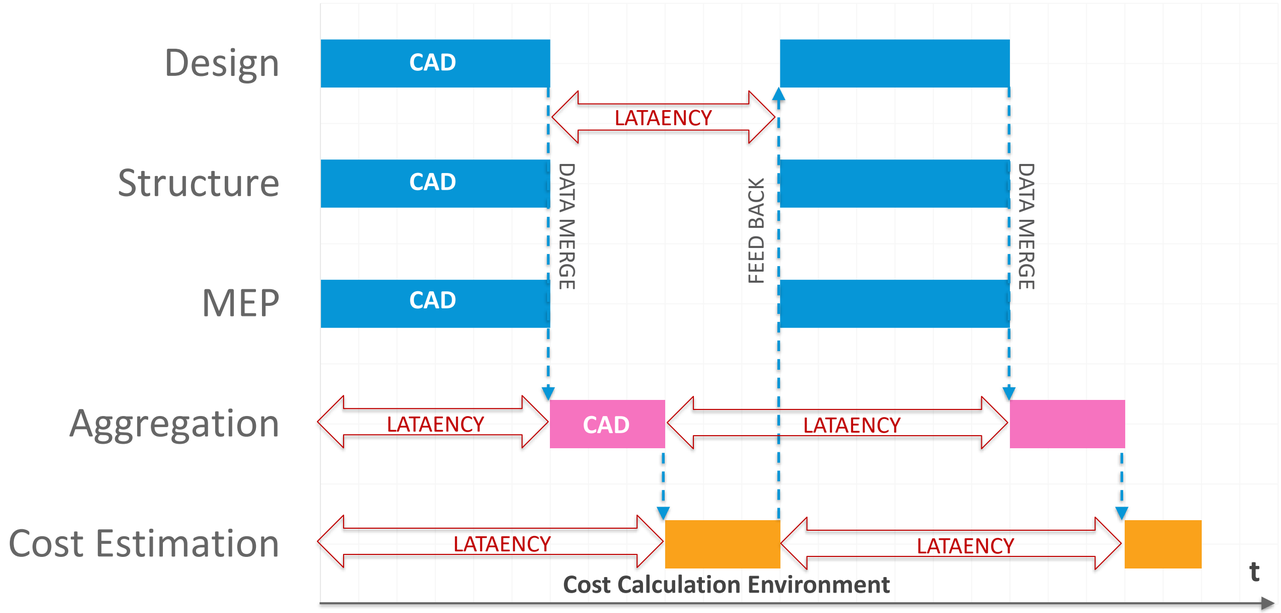
How Changes in Data Management Process by BIM and PaaS ?

Legacy Process of Data Management

About Legacy Process

- Project data of each section such as Design, Structure, MEP, etc. had been only unified at break of each design phases, so adjustment of design beyond section had been secured by communication and strict workflow.
- Project data are always synchronous, so some process has to wait to use the data until the other process finish using them. **As the result, the throughput decrease.**

DATA WATERFALL

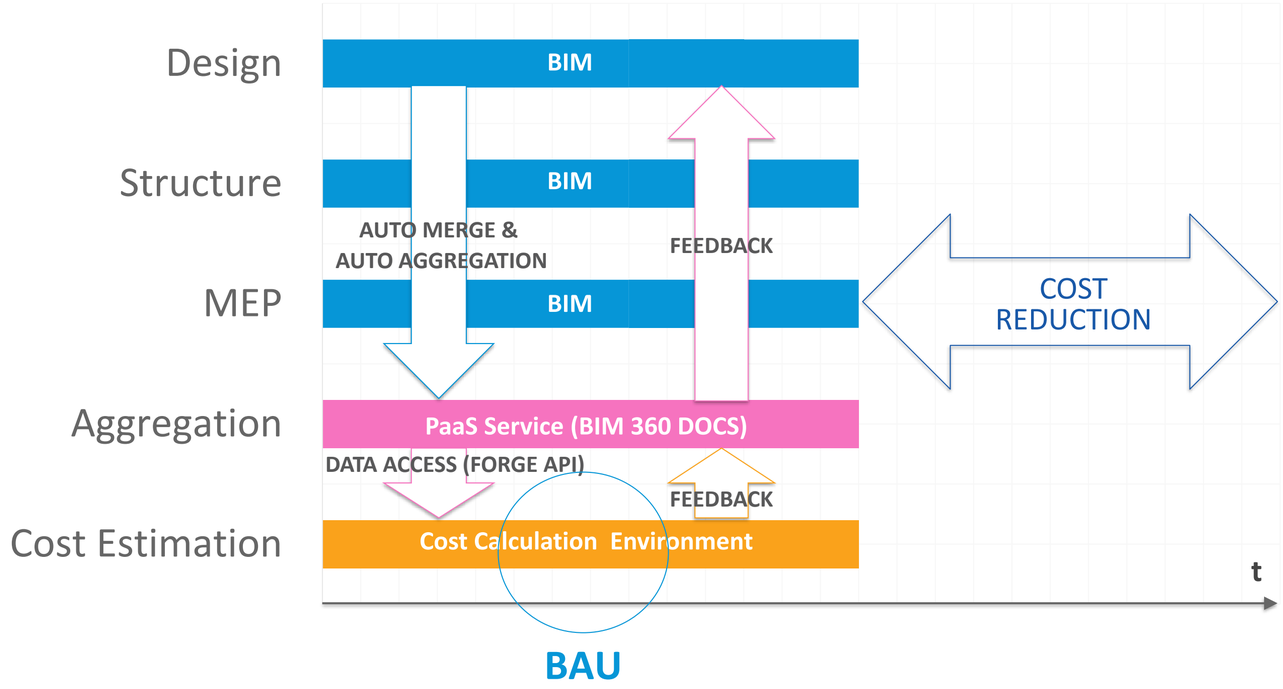


Current Process of Data Management

About Current Process

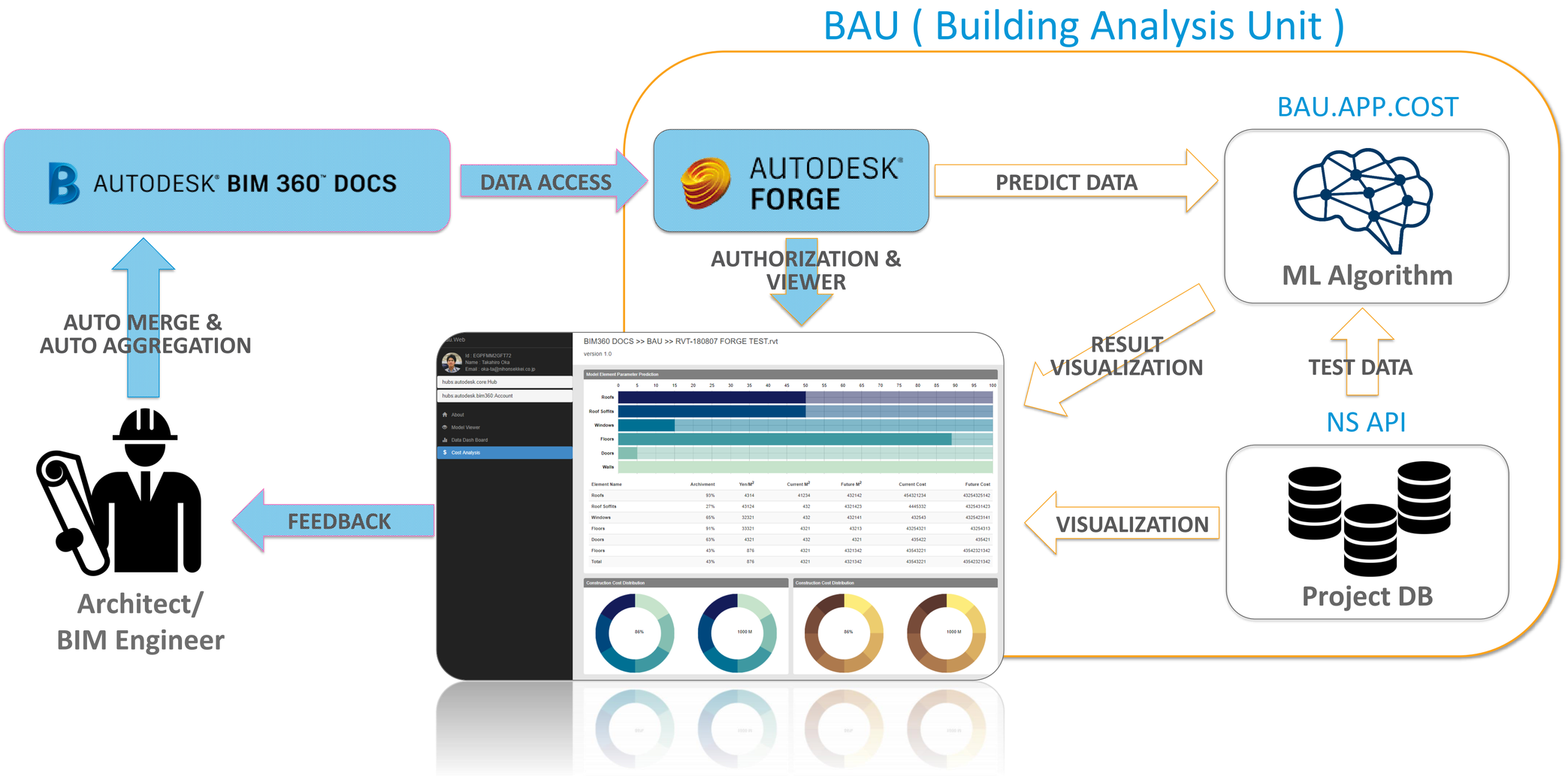
- Project data are uploaded to Autodesk cloud service like BIM 360 DOCS every specified time and task and merge them automatically.
- The project data can be accessed every time for adjustment of design beyond section and You can always do your own analysis on that data by your Forge Application.
- If other section 's process is not related to your tasks, you can process your task without waiting to finish of other process. which is like asynchronous operation. **As the result , the throughput increase.**

DATA WATERFALL



About Our Application by Forge

System Overview



Movie 1

Adapted Technologies in Front-End

The following technologies are mainly adopted on BAU.WEB.

- **ASP.NET Core** - Open Source Framework to design Cloud Base Application.
- **WebPack** - Module Handler to manage JS Library.
- **React** - JS library for building user interface as SPA.
- **Redux** - Redux is a predictable state container for JavaScript apps.
- **React-hot loader** - Tweak React components in real time for development.
- **Bootstrap** - Free and open-source front-end framework for designing web applications.
- **Babel** - Convert ES 2015+ code into a backwards compatible version of JavaScript.
- **Axios** - Promise based HTTP communication library.

About Forge APIs

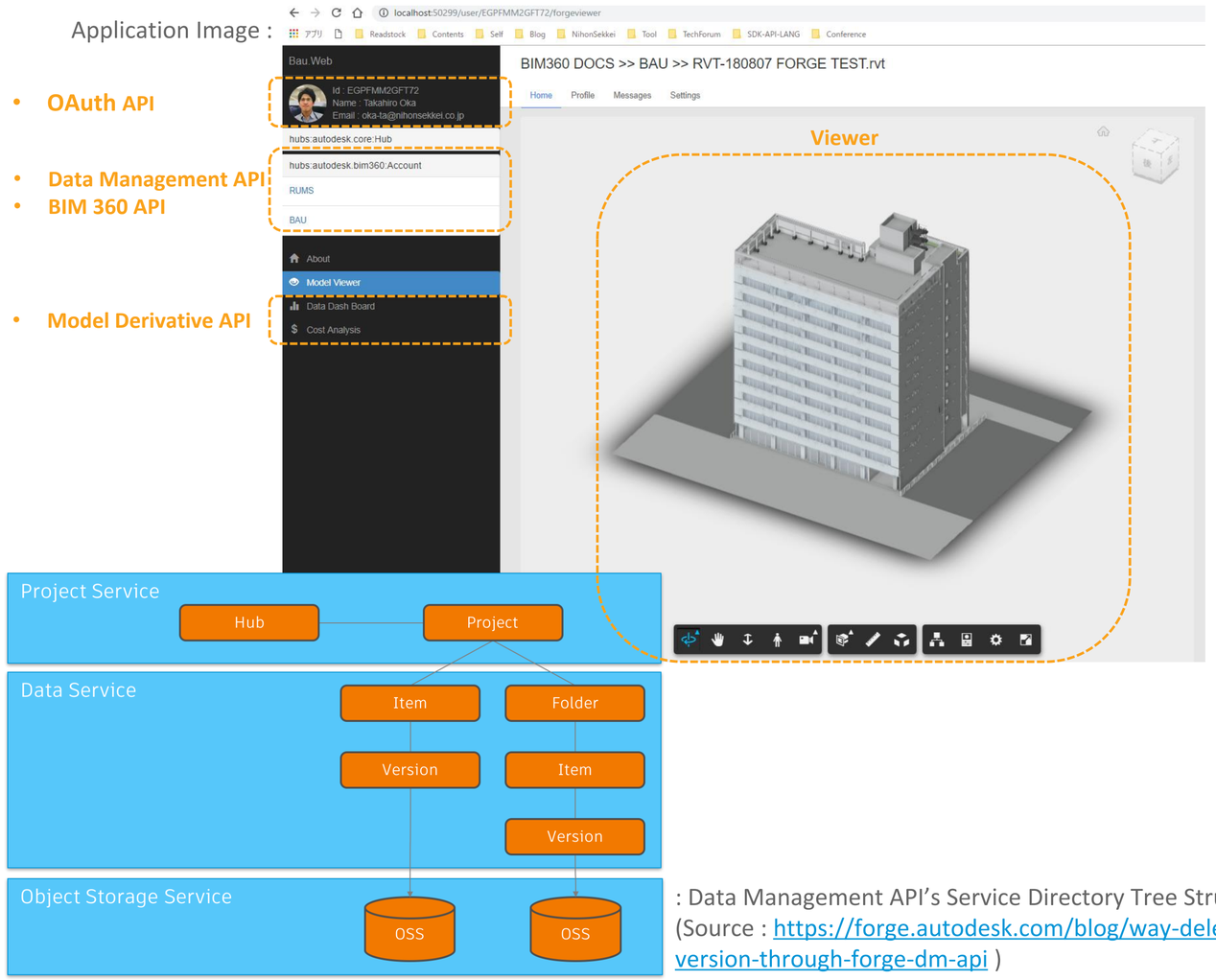
Forge is a set of web services has been provided by Autodesk, and It can be used from any application that can have environment to use HTTP protocol.

Currently, provides services are :

- **OAuth**
- **Data Management API**
- Design Automation API
- **Model Derivative API**
- **Viewer**
- Reality Capture API
- **BIM 360 API**
- Webhooks API

Application Image :

- **OAuth API**
- **Data Management API**
- **BIM 360 API**
- **Model Derivative API**



: Data Management API's Service Directory Tree Structure (Source : https://forge.autodesk.com/blog/way-delete-file-version-through-forge-dm-api)

How get objects of 3D Model?

WHAT API NEED?

After you get access_token, you need some API request.

Data Management API

1. https://developer.api.autodesk.com/project/v1/hubs
2. https://developer.api.autodesk.com/project/v1/hubs/{hub_id}/projects
3. https://developer.api.autodesk.com/project/v1/hubs/{hub_id}/projects/{project_id}/topFolder
4. https://developer.api.autodesk.com/data/v1/projects/{project_id}/folders/{folder_id}/contents
5. https://developer.api.autodesk.com/data/v1/projects/{project_id}/items/{item_id}

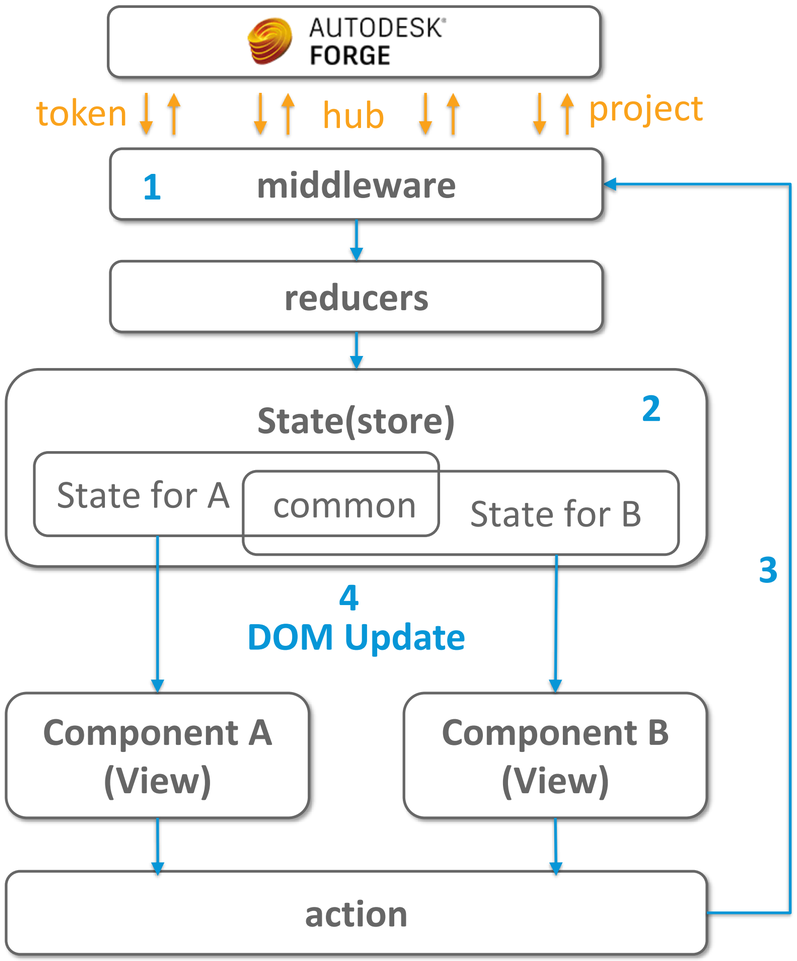
Model Derivative API

6. https://developer.api.autodesk.com/modelderivative/v2/designdata/{urn}/metadata
7. https://developer.api.autodesk.com/modelderivative/v2/designdata/{urn}/metadata/{guid}

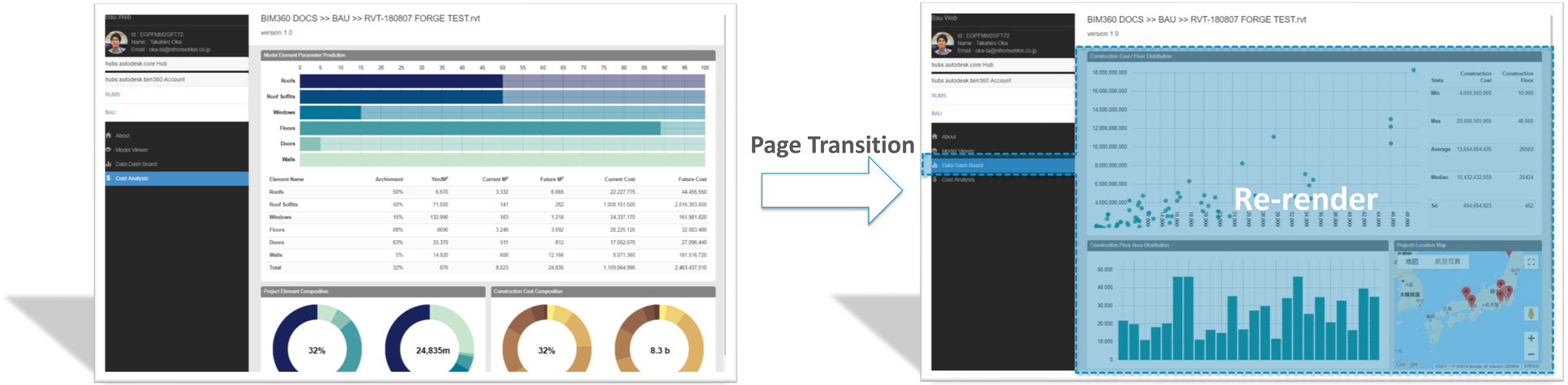
3D model object data :

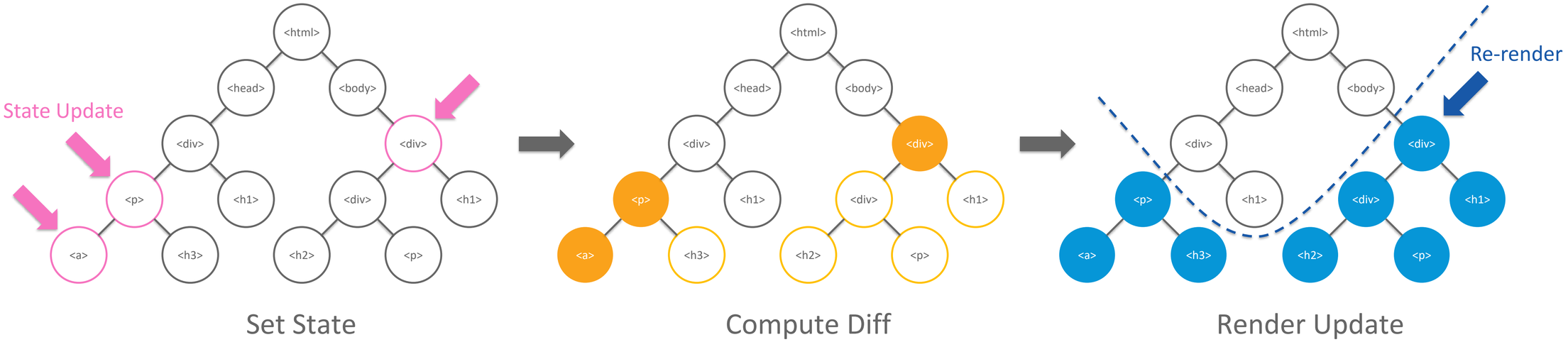
Why Did I adopt React and Redux for Forge App?

1. When we use Forge API for Web App, many requests are generated. Therefore we need to call API from client side to ensure usability as Single Page Application (hereinafter, this is called “SPA") **React is one of the User Interface Libraries for SPA.**
2. If responded data is stored even once by some action in the view, the data can be read from the other view, so **you can short cut view transition time** except re-rendering time.
3. In React and Redux Framework, state are strongly related to each DOM. Therefore if state is updated , DOM is automatically re-rendered. By this process which is one way data flow, **we can make responsive user interface easily** without complex scraping coding such as jQuery.
4. React is adopt **Virtual DOM concept which accelerate DOM tree re-rendering.**



About Virtual DOM





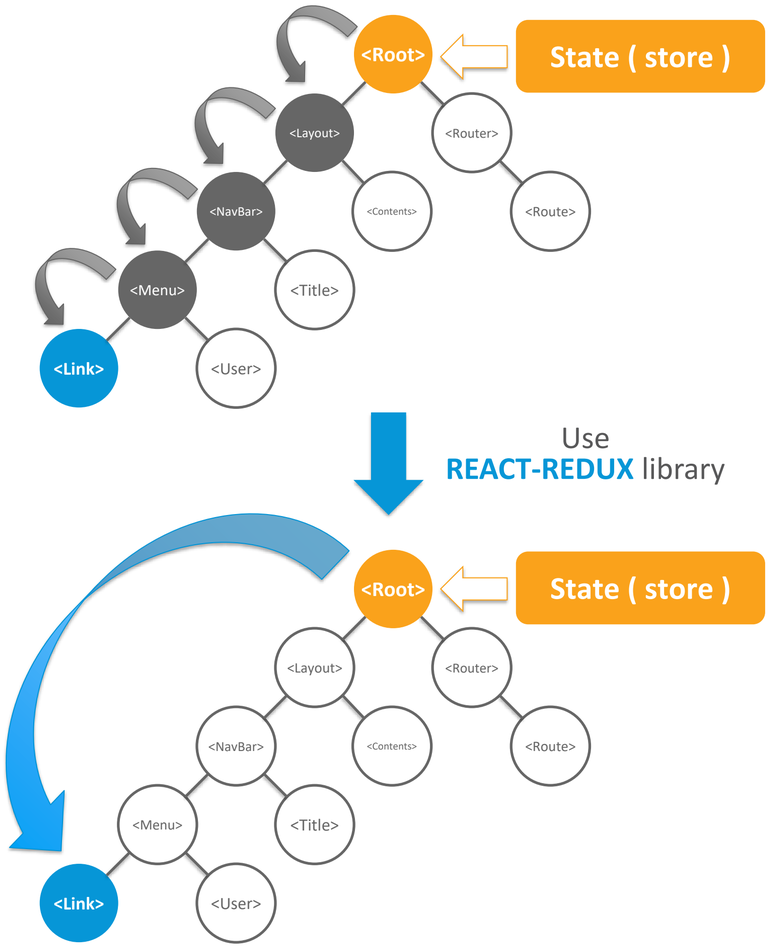
About React-Redux Library

WHAT IS REACT-REDUX ?

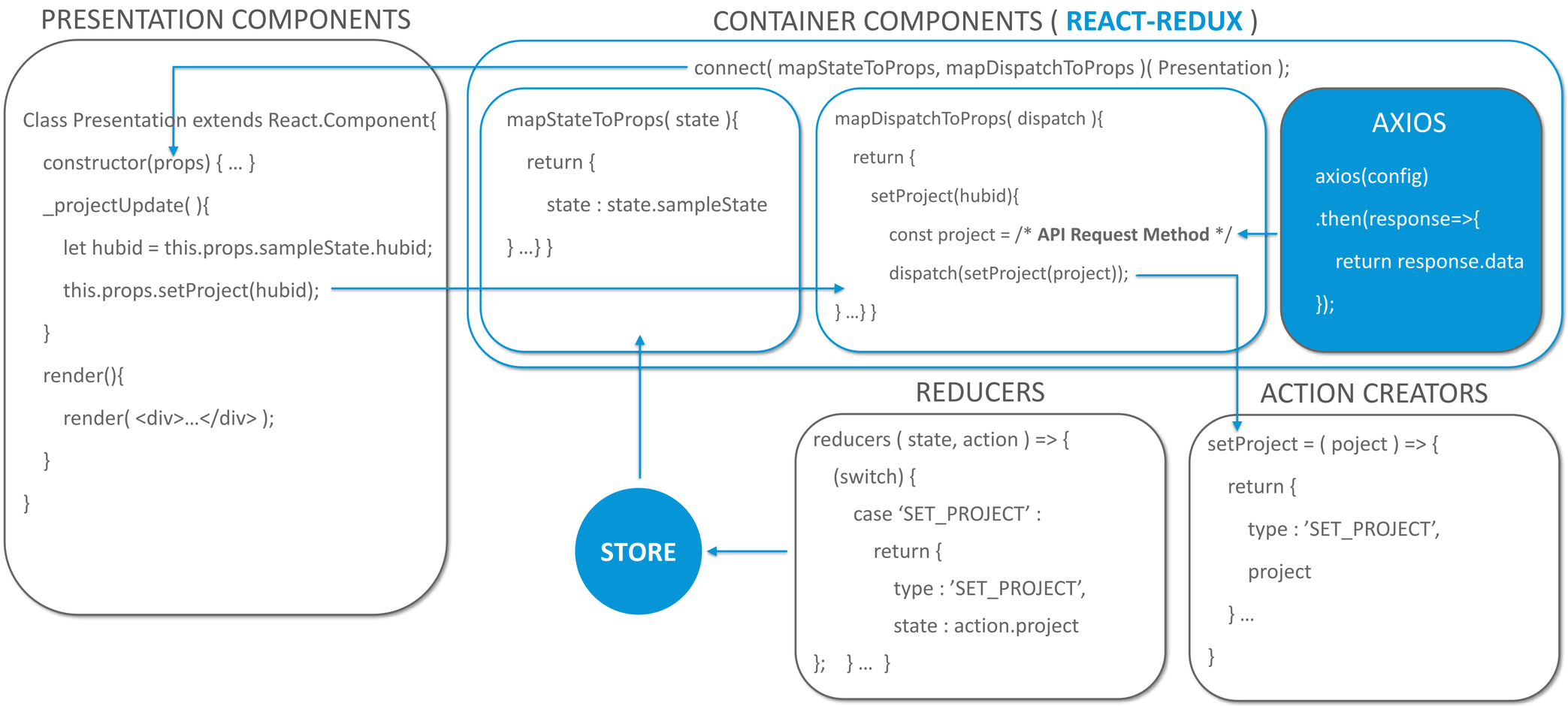
React-Redux is official React bindings for Redux. It support your developing of React and Redux application by wrapping some order. This library's feature are

1. Component(View) get application state (store) without data relay from parents.
2. It separate dispatch method, which is update your application state(store), from Component(View).

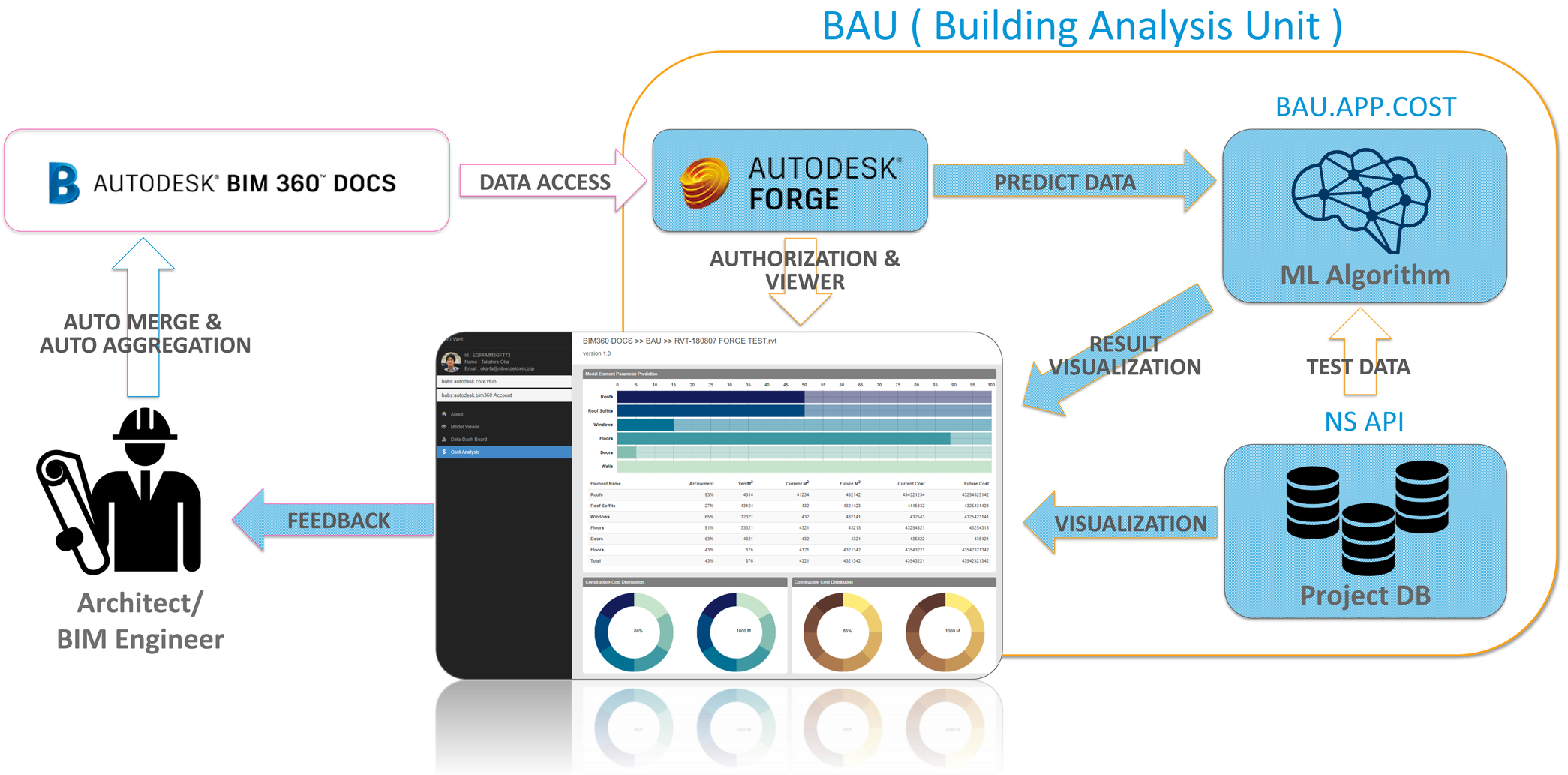
As the result, you can develop your application while keeping your code clean.



React-Redux Lifecycle with API Request



System Overview



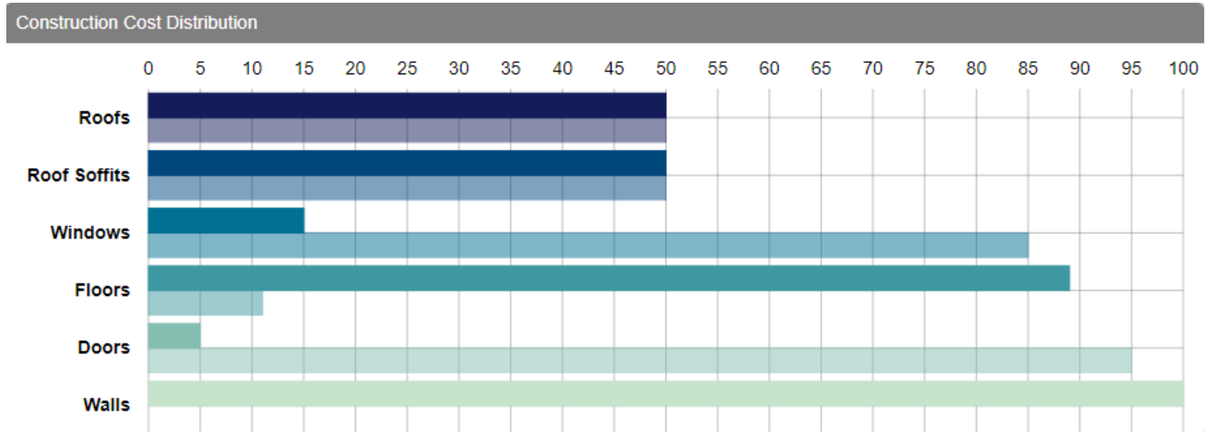
Movie 2

Visualization Tools (React-vis)

WHAT IS REACT-VIS ?

“react-vis” is open source library, which have been developing by *uber*, for data visualization like d3.js in React app.

Created histogram can set animation and be customized freely, so It is suitable for you If you want to make dashboard application like this.




```
const greenData = [{y: "Walls", x: 100, color:0},...{y: "Roofs", x: 50, color:5}];
const blueData = [{y: "Walls", x: 0, color:0, opacity: 0.5},...,
                  {y: "Roofs", x: 50, opacity: 0.5, color:5}];

<FlexibleXYPlot
    yType="ordinal"
    height={300}
    colorType="category"
    colorDomain={[0,1,2,3,4,5]}
    colorRange={["#c6e3cb","#84beb1","#3e98a1","#007093","#00477d","#141c59"]}
    margin={{left: 110, right: 16, top:35, bottom:0}} >
    <VerticalGridLines />
    <HorizontalGridLines />
    <XAxis align={{vertical: 'top', horizontal: 'left'}} top={0} />
    <YAxis />
    <HorizontalBarSeries data={greenData} animation />
    <HorizontalBarSeries data={blueData} animation />
</FlexibleXYPlot>
```

Visualization Tools (React-google-maps)

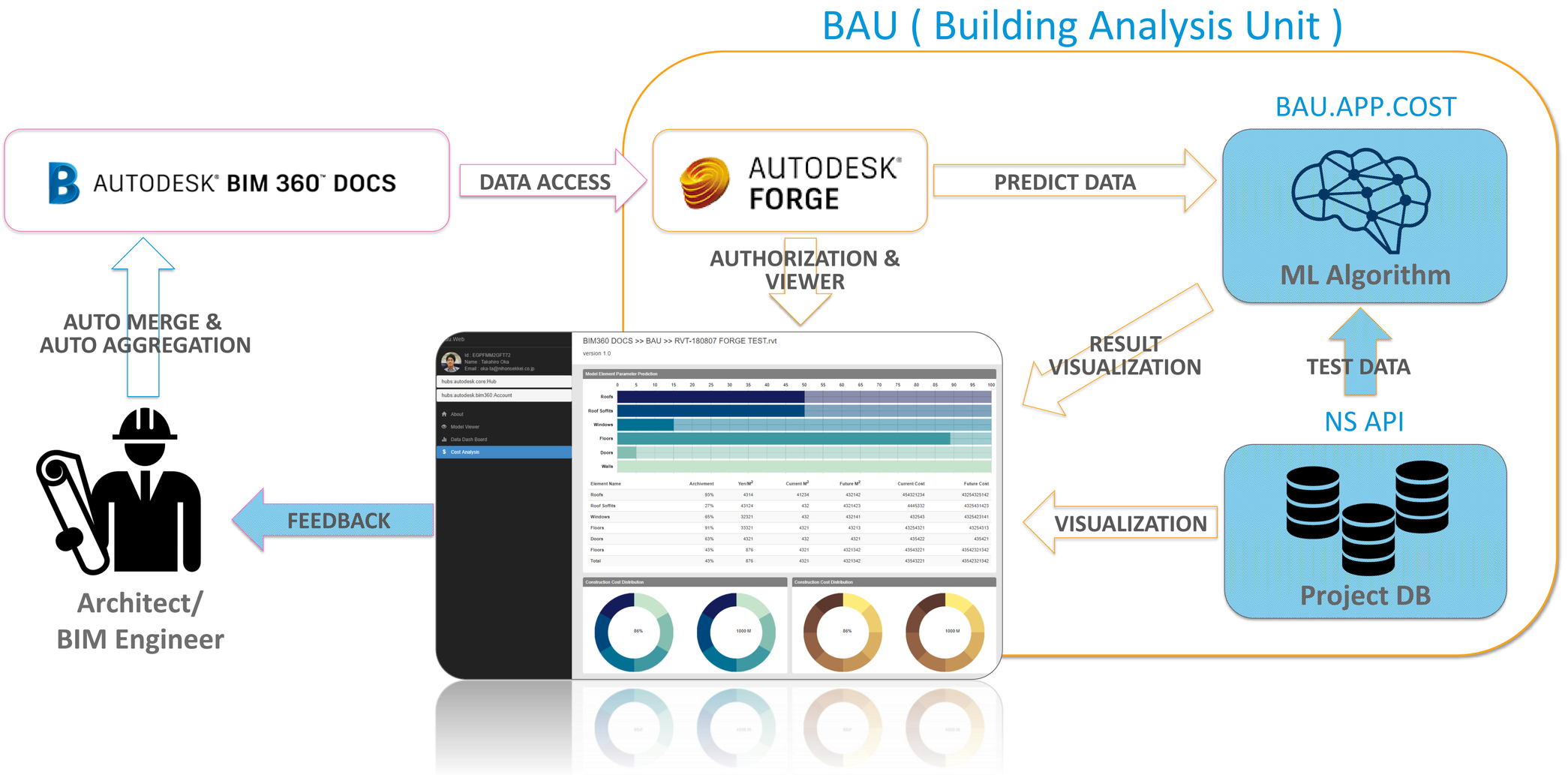
```
export const ProjectsMap = (props) =>{

    return (
        <Map
        isMarkerShown = {true}
        data = {[{lat:35.243, lng:139.05},
                 {lat:34.963, lng:138.27},
                 {lat:35.646, lng:139.90},
                 {lat:35.544, lng:134.82}]}
        googleMapURL="https://maps.googleapis.com/maps/api/js?key=<API Key>"
        loadingElement={<div style={{ height: `100%` }} />}
        containerElement={<div style={{ height: `100%` }} />}
        mapElement={<div style={{ height: `100%` }} />}
        />
    );
}

const Map = withScriptjs(withGoogleMap((props) =>
    <GoogleMap
    defaultZoom = {5}
    defaultCenter={{lat:34.665108, lng:135.000000}}>
        {props.isMarkerShown &&
        props.data.map(location=><Marker position={location}/>)}
    </GoogleMap>
))
```

WHAT IS “REACT-GOOGLE-MAPS” ?

“react-google-maps” is open source library, which wrapped google maps javascript API, for displaying map in React app.

About Back-End (Business Logic)

System Overview

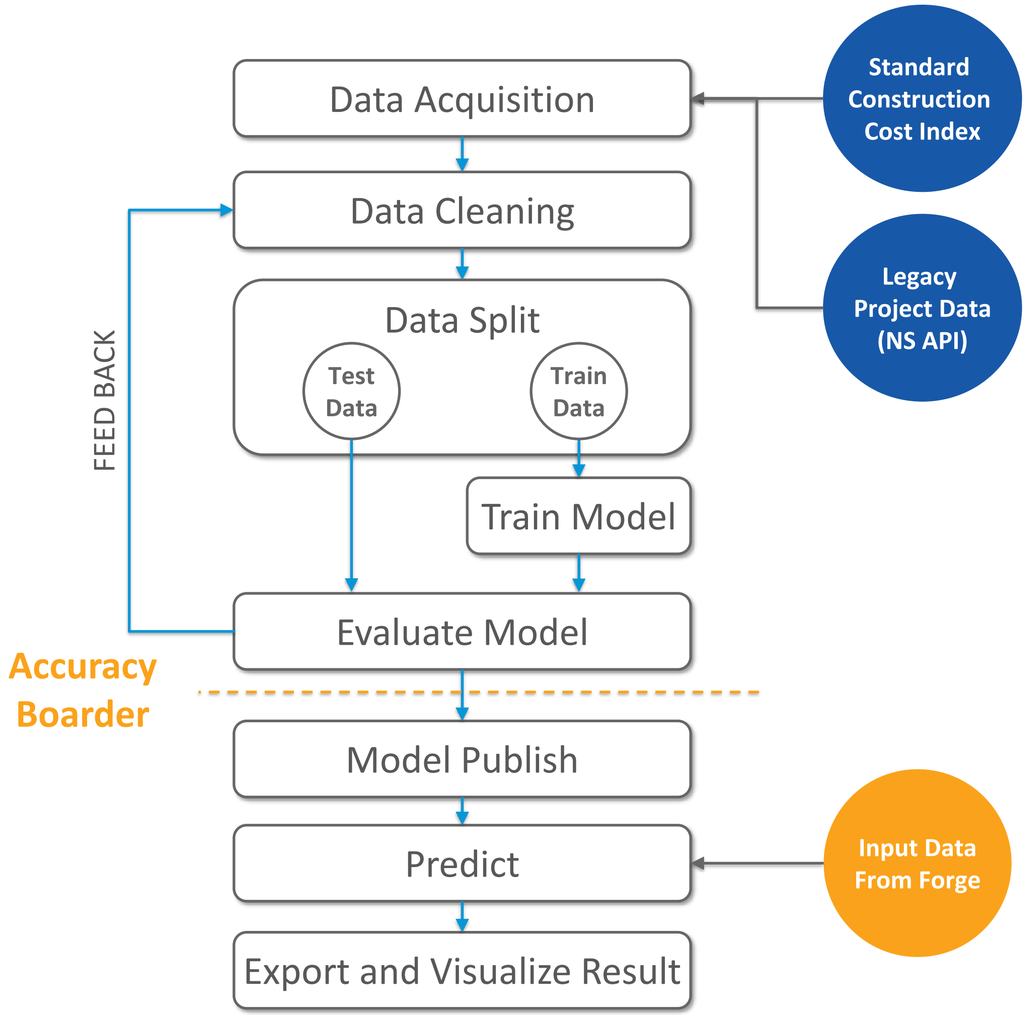


Adapted Technologies

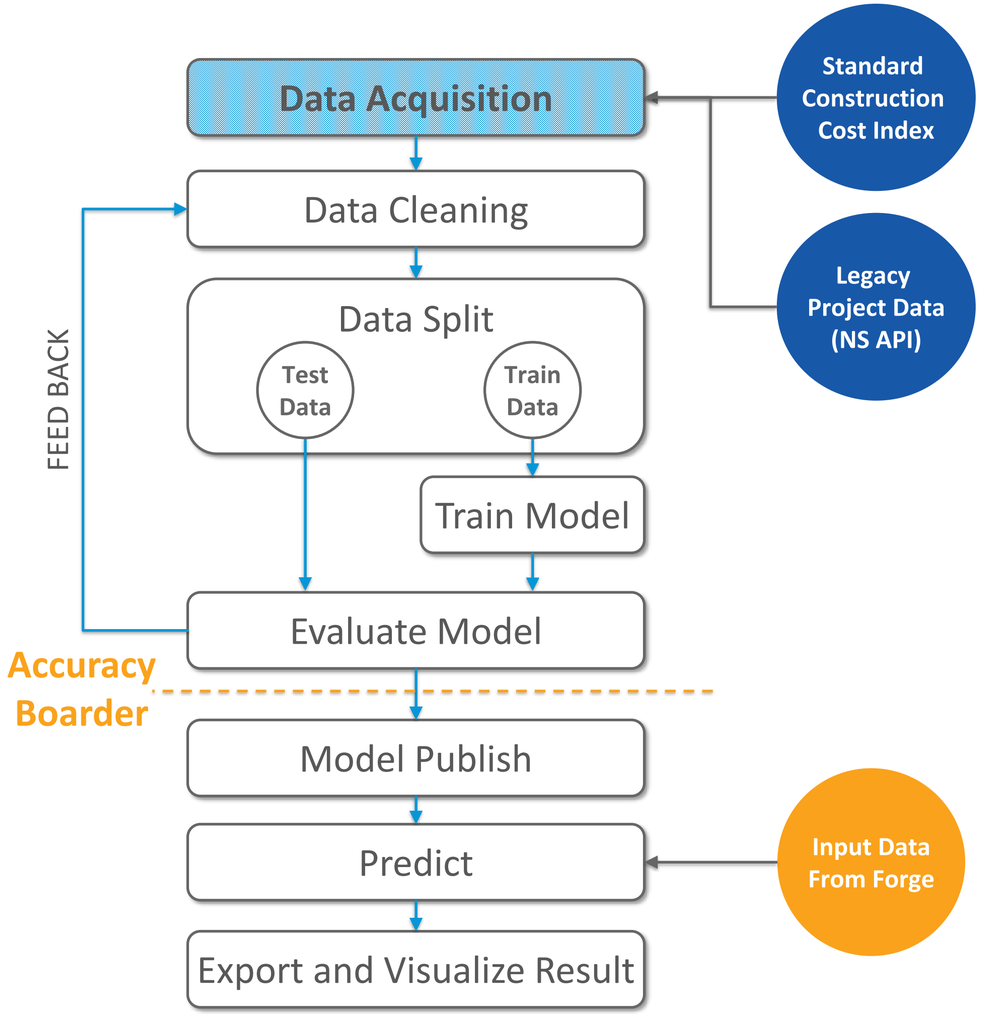
The following technologies are mainly adopted on BAU.APP.COST

- **Flask** - Flask is a micro web framework written in Python.
- **scikit-learn** - It provides simple and efficient tools for data mining and data anlysis.
- **Jupyter Notebook** - It is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text.
- **Matplotlib** - It is a Python 2D plotting library.
- **Numpy** - NumPy is the fundamental package for scientific computing with Python.

Machine Learning Process

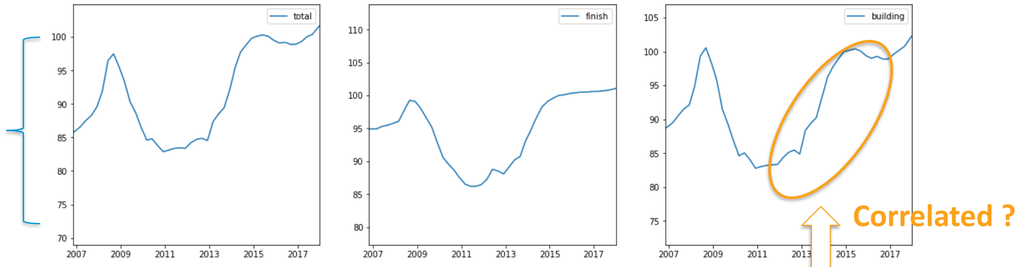


Data Acquisition

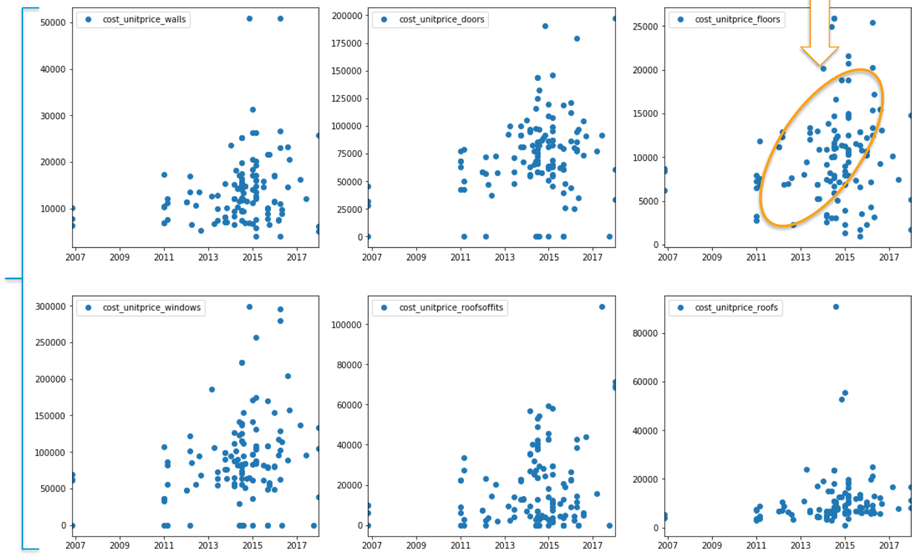


How decide data?

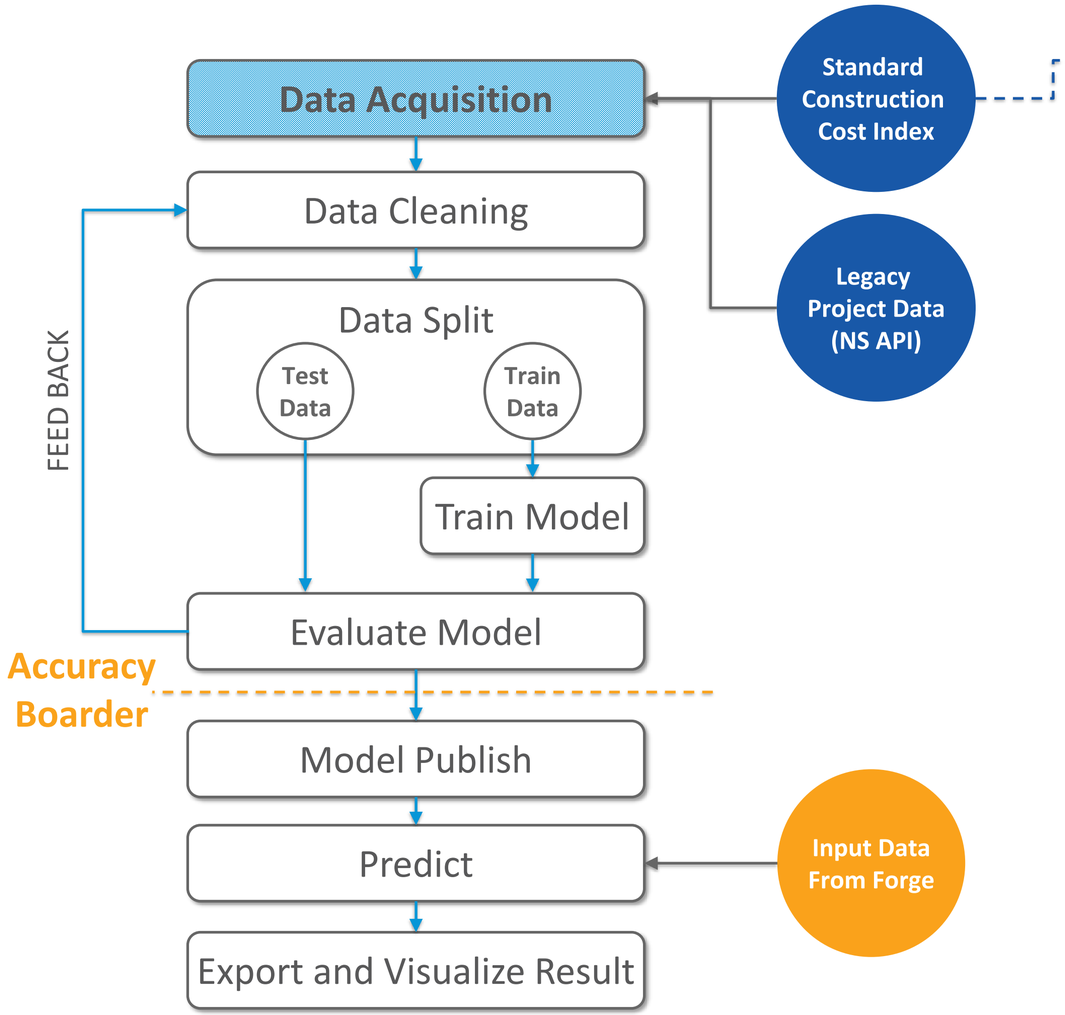
Price Index By Year



Finish's Unit Price Distribution by Year (Walls, Doors, Floors, Roofs, Roof Soffits, Windows)

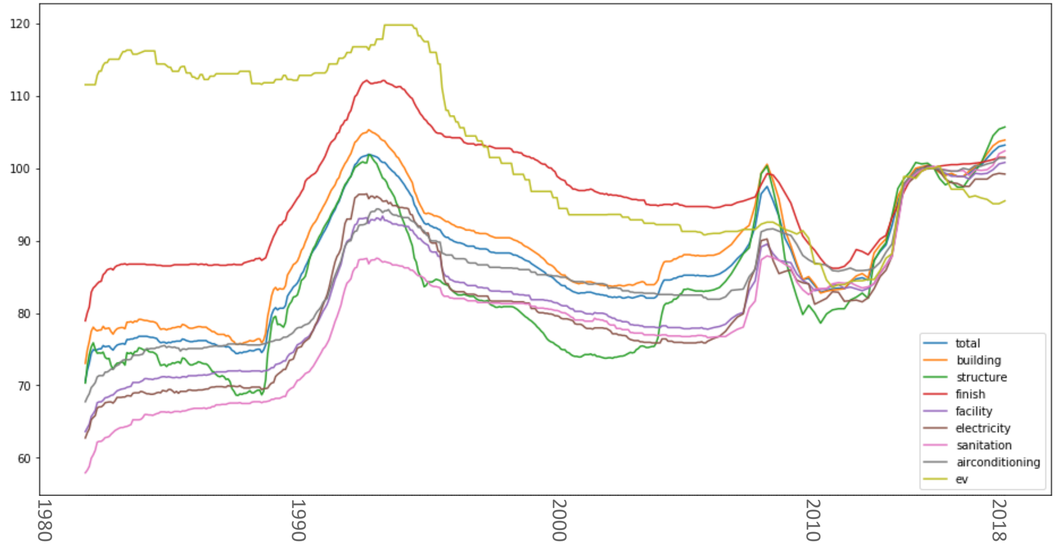


Data Acquisition



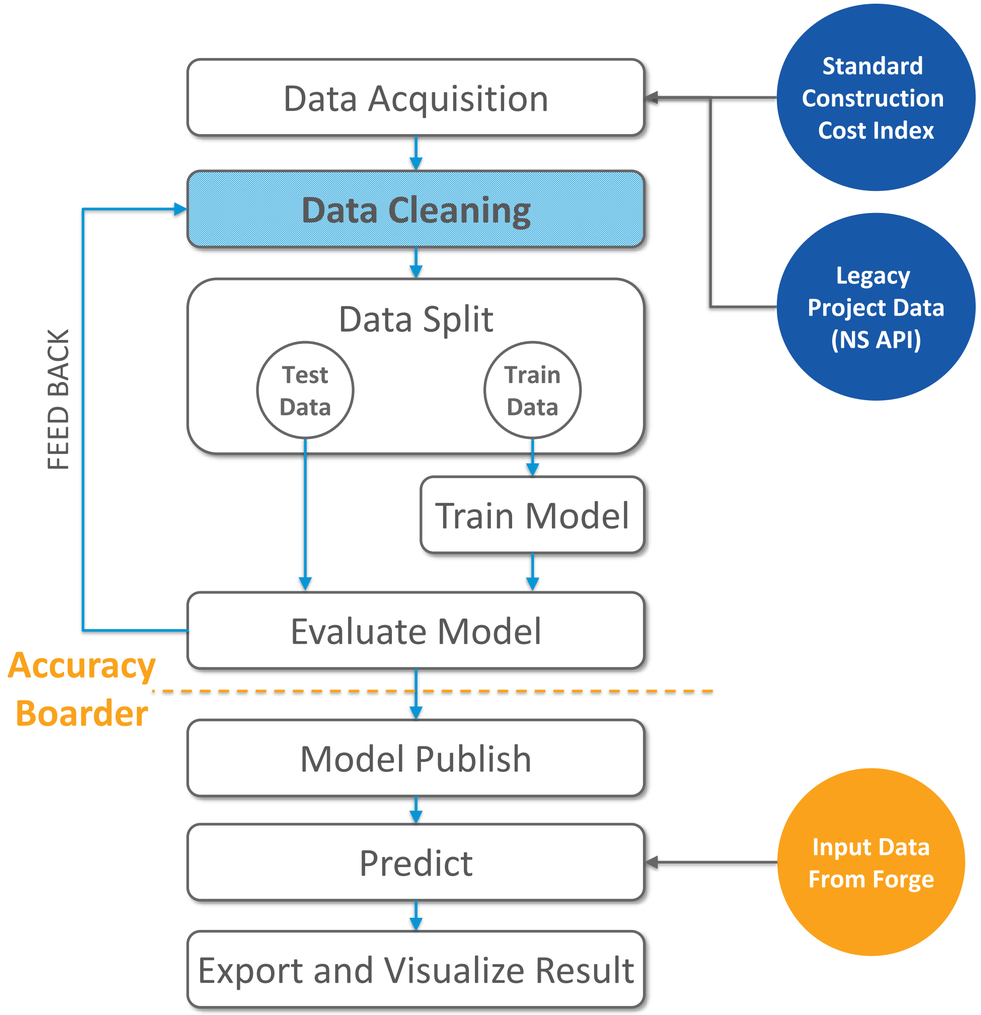
Standard Construction Cost Index (ja : 標準建築費指数)

This is one of the construction price index that reported by *Kensetu-Kogyo-Kenkyukai.* It reported value of each building element such as Finish, Structure, etc.



: Standard Construction Cost Index in Tokyo (average: Tokyo 2015 = 100)
(Source : https://www.yoi-kensetsu.com/dantai/info.php?did=111)

Data Cleaning



How clean the data?

	general-sideX	general-sideY	general-spanX	general-spanY
20	0	0.00	0	0
21	117.25	65.45	10.8	7.2
22	332.6	95.60	9	9
23	0	0.00	0	0
24		62.00		
25	144	80.00	12	8
26	46.78	42.00	6	5.3
27	50.4	18.22	7.2	12.25
28	135.45	69.40	9	9
29	62.23	37.16	6.4	19.75

COMPLETION & DROP NAN

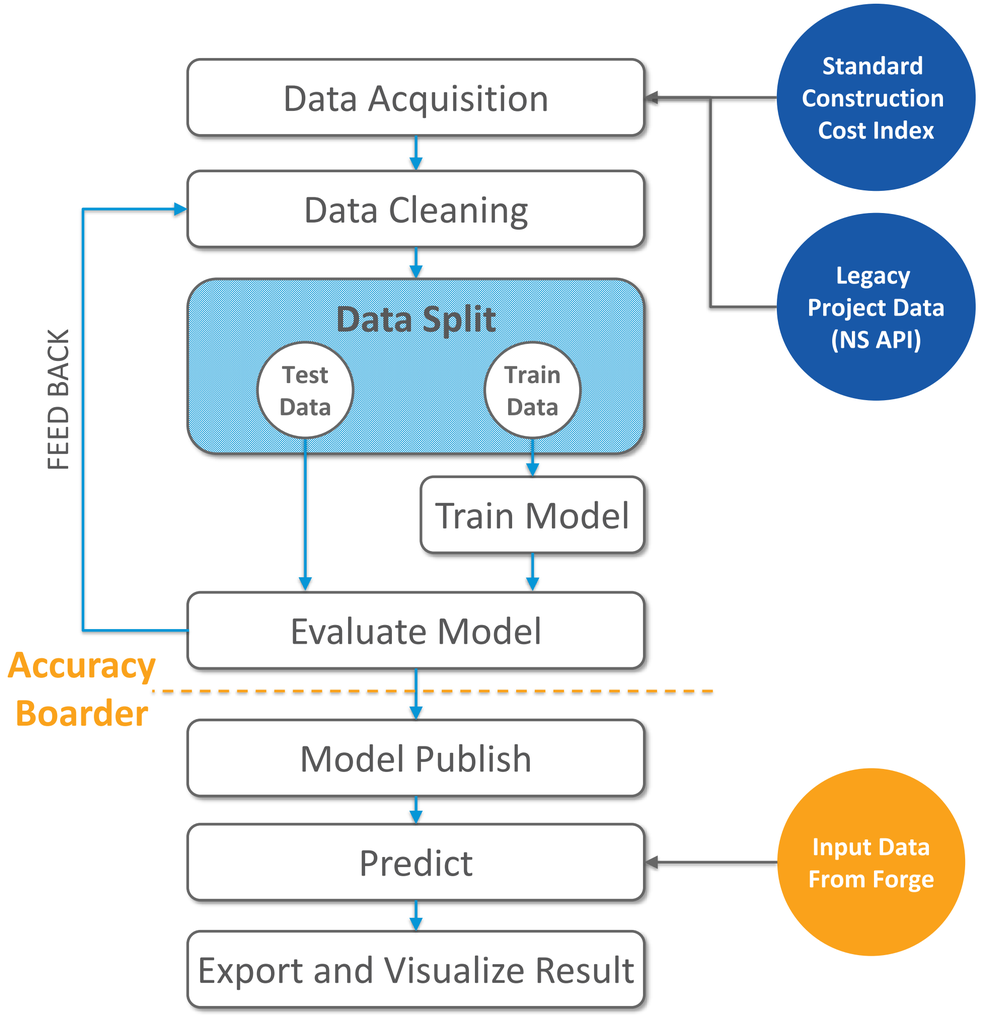
	general-sideX	general-sideY	general-spanX	general-spanY
20	126.958571	58.261429	8.628571	10.071429
21	117.250000	65.450000	10.800000	7.200000
22	332.600000	95.600000	9.000000	9.000000
23	126.958571	58.261429	8.628571	10.071429
25	144.000000	80.000000	12.000000	8.000000
26	46.780000	42.000000	6.000000	5.300000
27	50.400000	18.220000	7.200000	12.250000
28	135.450000	69.400000	9.000000	9.000000
29	62.230000	37.160000	6.400000	19.750000

NORMALIZATION

	general-sideX	general-sideY	general-spanX	general-spanY
20	0.000000	-6.459852e-16	-9.685332e-16	0.000000
21	-0.119681	3.267721e-01	1.183940e+00	-0.738617
22	2.535013	1.697306e+00	2.025161e-01	-0.275603
23	0.000000	-6.459852e-16	-9.685332e-16	0.000000
25	0.210076	9.881738e-01	1.838223e+00	-0.532833
26	-0.988389	-7.391984e-01	-1.433191e+00	-1.227353
27	-0.943764	-1.820170e+00	-7.789082e-01	0.560393
28	0.104677	5.063279e-01	2.025161e-01	-0.275603
29	-0.797931	-9.592111e-01	-1.215097e+00	2.489615

Adjust data for improving accuracy of machine learning model.

Data Split



How split the data ?

	general-sideX	general-sideY	general-spanX	general-spanY
20	0.000000	-6.459852e-16	-9.685332e-16	0.000000
21	-0.119681	3.267721e-01	1.183940e+00	-0.738617
22	2.535013	1.697306e+00	2.025161e-01	-0.275603
23	0.000000	-6.459852e-16	-9.685332e-16	0.000000
25	0.210076	9.881738e-01	1.838223e+00	-0.532833
26	-0.988389	-7.391984e-01	-1.433191e+00	-1.227353
27	-0.943764	-1.820170e+00	-7.789082e-01	0.560393
28	0.104677	5.063279e-01	2.025161e-01	-0.275603
29	-0.797931	-9.592111e-01	-1.215097e+00	2.489615

70%

	general-sideX	general-sideY	general-spanX
23	0.000000	-6.459852e-16	-9.685332e-16
20	0.000000	-6.459852e-16	-9.685332e-16
25	0.210076	9.881738e-01	1.838223e+00
26	-0.988389	-7.391984e-01	-1.433191e+00
29	-0.797931	-9.592111e-01	-1.215097e+00
21	-0.119681	3.267721e-01	1.183940e+00

TRAINING INPUT

	general-spanY
23	0.000000
20	0.000000
25	-0.532833
26	-1.227353
29	2.489615
21	-0.738617

TRAINING OUTPUT

30%

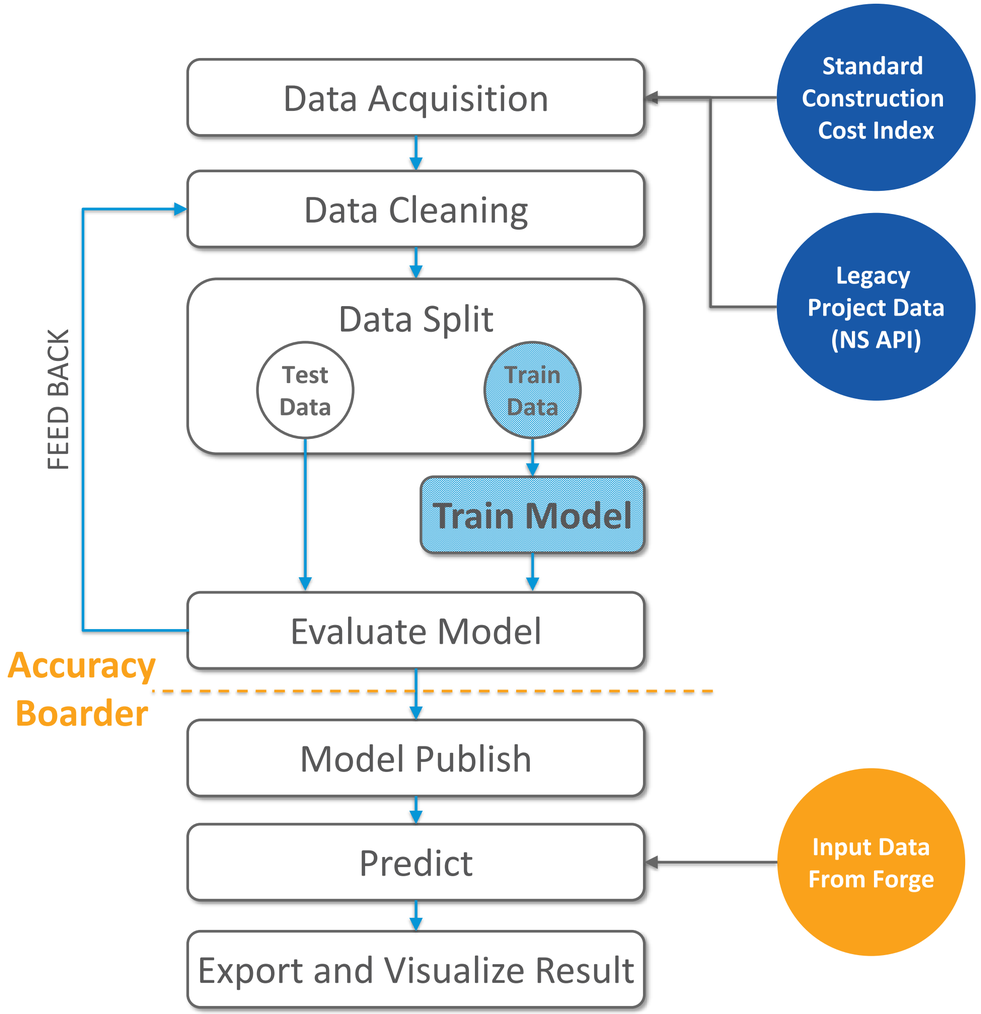
	general-sideX	general-sideY	general-spanX
28	0.104677	0.506328	0.202516
22	2.535013	1.697306	0.202516
27	-0.943764	-1.820170	-0.778908

TEST INPUT

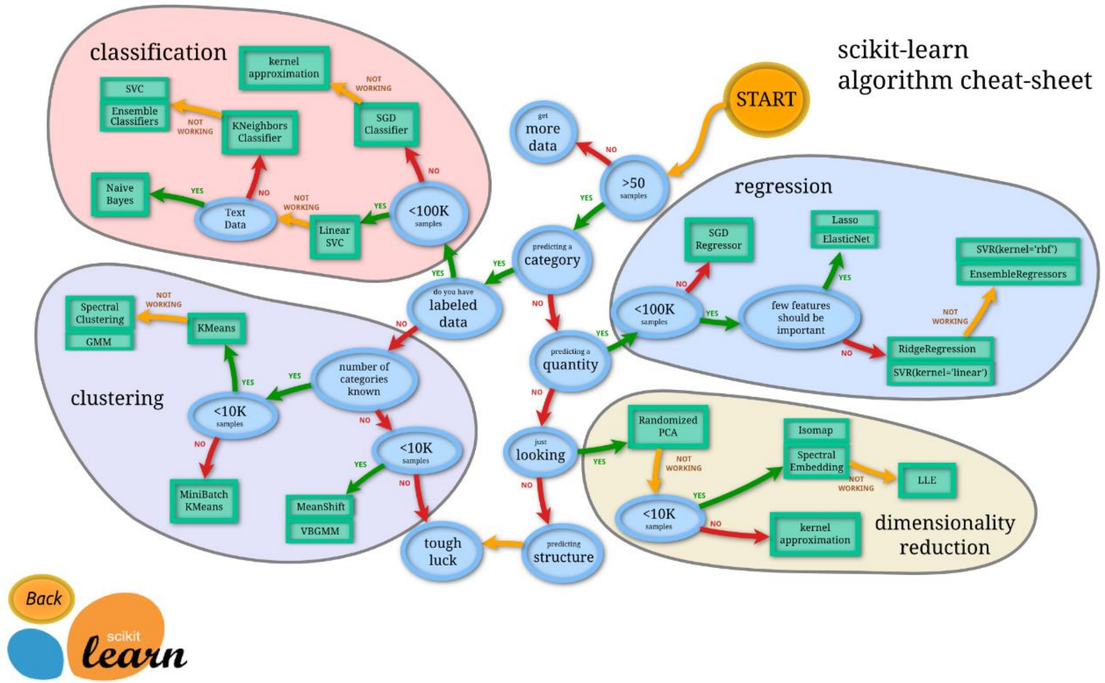
	general-spanY
28	-0.275603
22	-0.275603
27	0.560393

TEST OUTPUT

Train Model

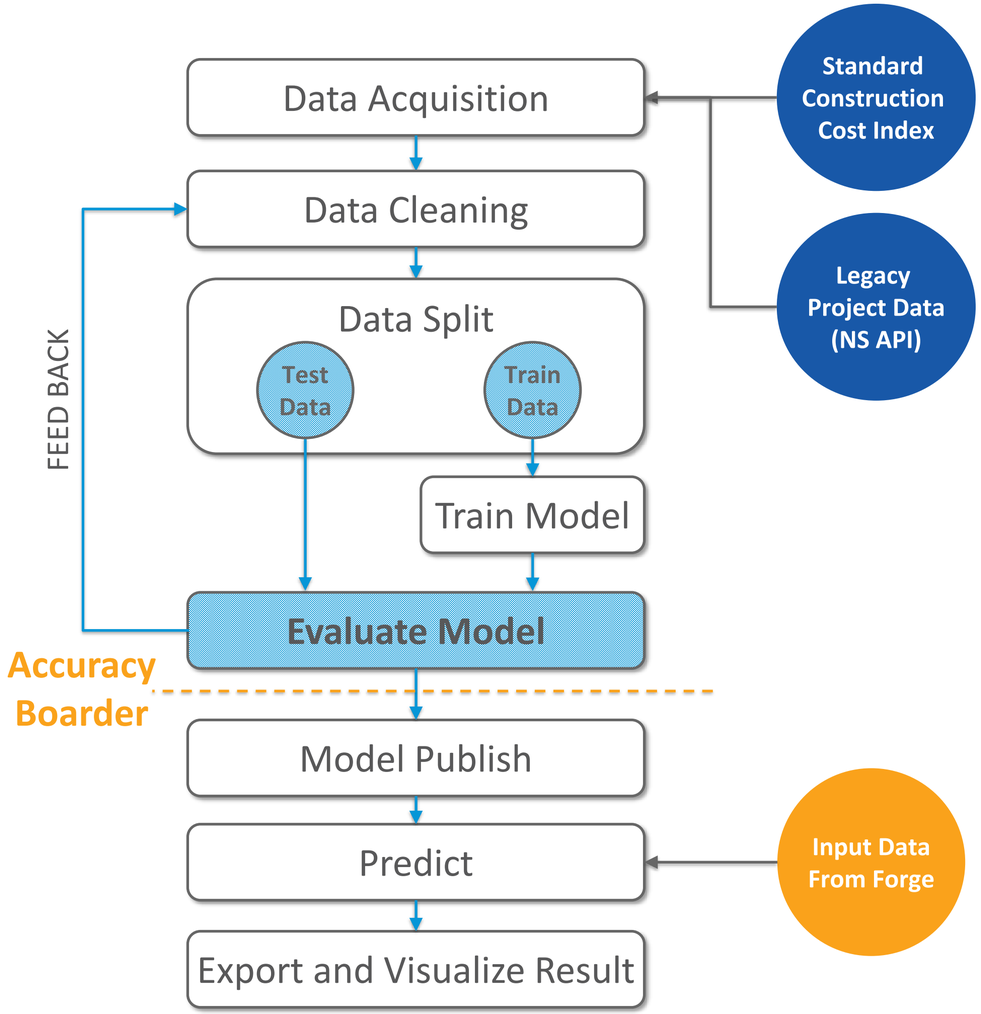


Which Algorithm is needed?

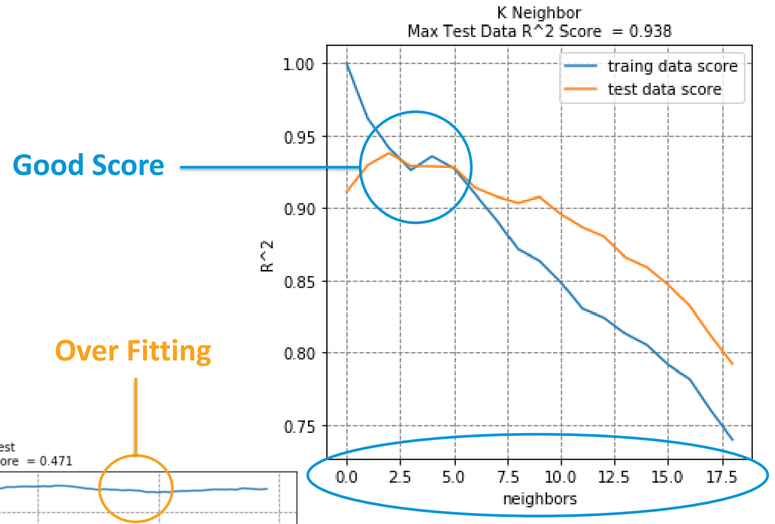


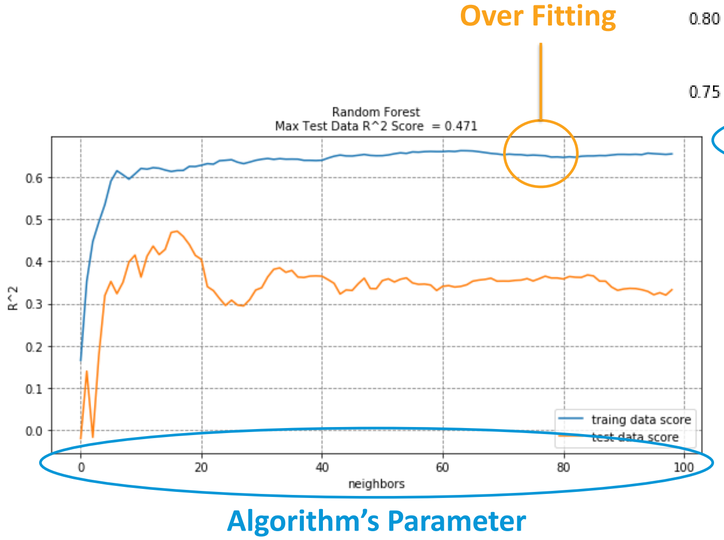
: Choosing the right estimator
(Source : https://scikit-learn.org/stable/tutorial/machine_learning_map/)

Evaluate Model

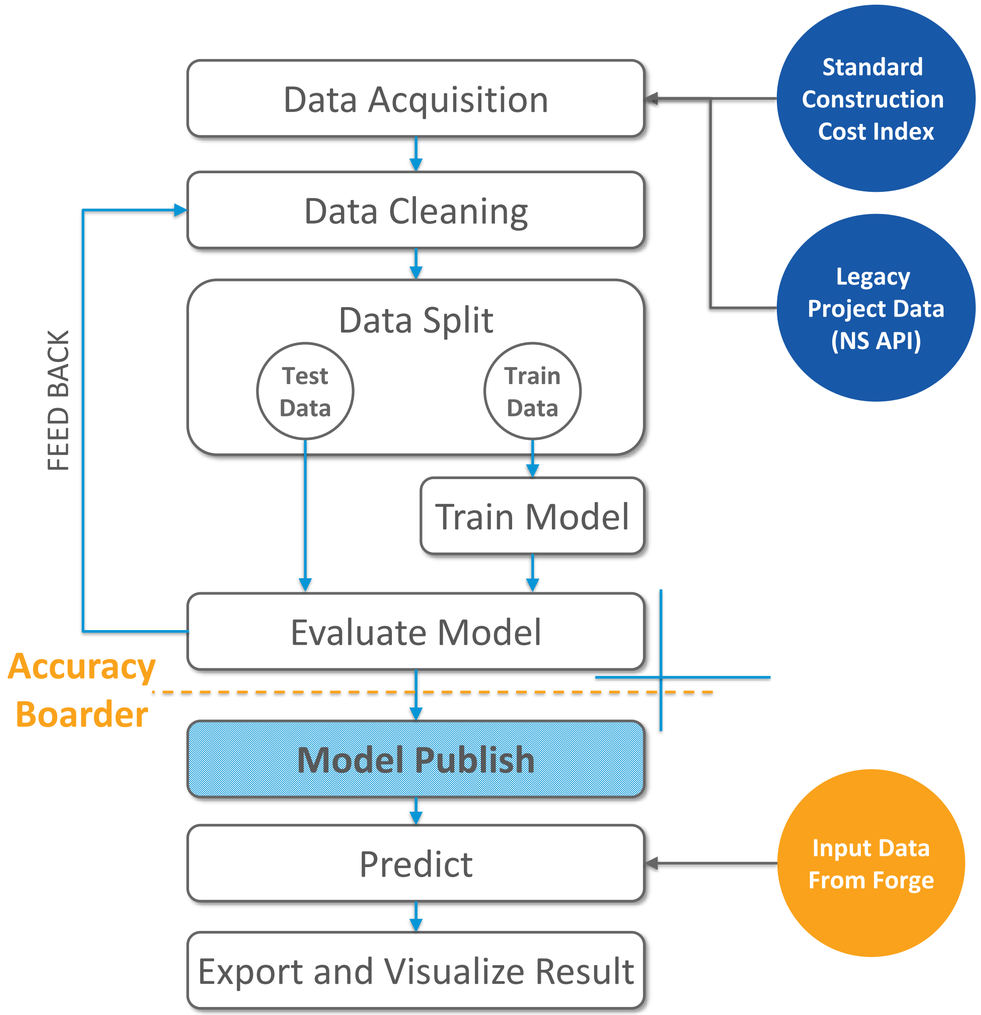


How Evaluate Model ?





Model Publish



How Publish Model ?

Publish Model

Regenerate Model

Publish Model as Web API

Call API from your system

Data Analysis Environment (Jupyter Notebook)

```
# In Jupyter Notebook for data analysis
regr = RandomForestRegressor(random_state=0, n_estimators=10)
regr.fit(np.array(X_train), np.array(y_train))

joblib.dump(regr,'../pkl/regr_construction_cost_composition.pkl')
```

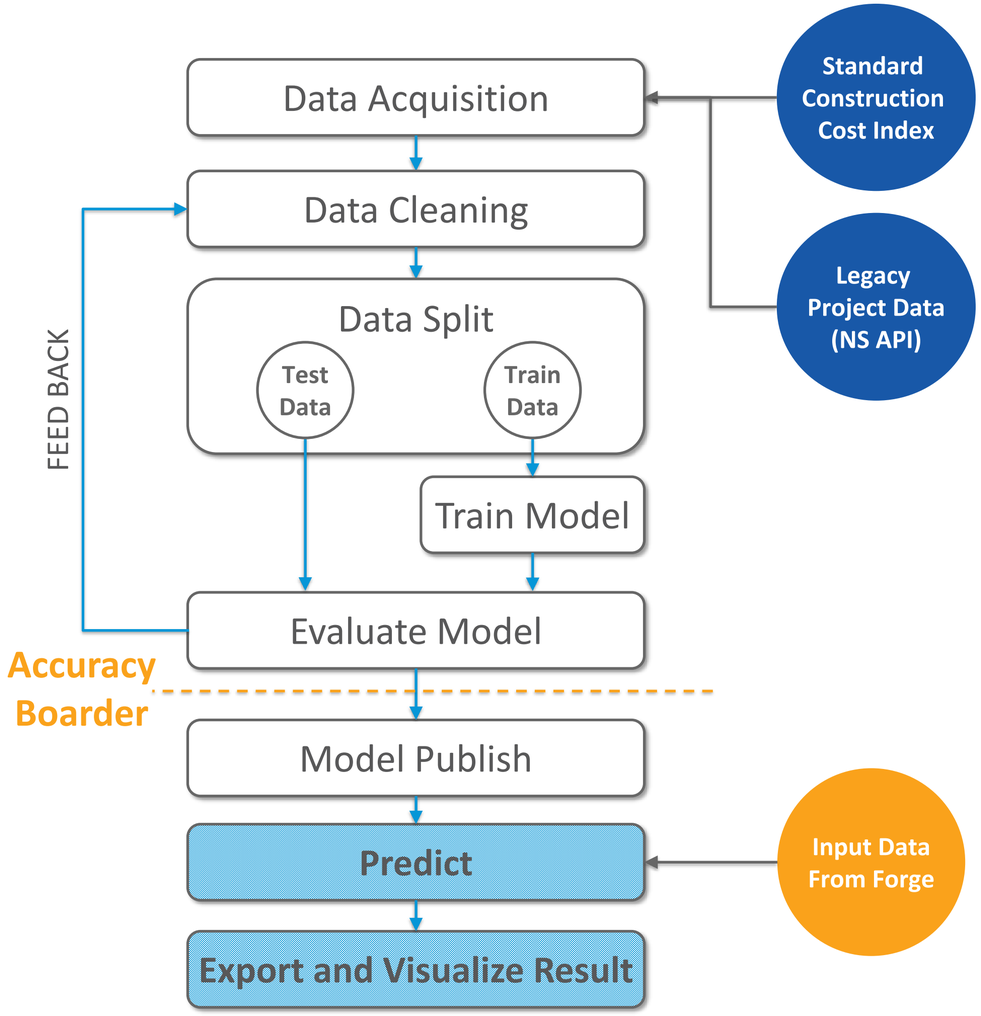
Application Server (Flask)

```
# Flask Application Server as API
@app.route("/cost-rf",method=["POST"])
def cost_rf():
            """
            Total Cost Prediction with Random Forest
            Algorithm of scikit-learn.
            """
            cost_total_prediction = 0
            response = None
            regr = joblib.load('./model/pkl/regr_construction_cost_composition.pkl')

        if request.method == 'POST':
                        area = float(request.form['total_floor_area'])
                        prediction = regr.predict(area)
                        response = {
                                    "cost_total_prediction" : cost_total_prediction.tolist(),
                                    "regr_coef"             : regr.coef_.tolist(),
                                    "regr_intercept"        : regr.intercept_.tolist()
                        }
            return jsonify(response)
```

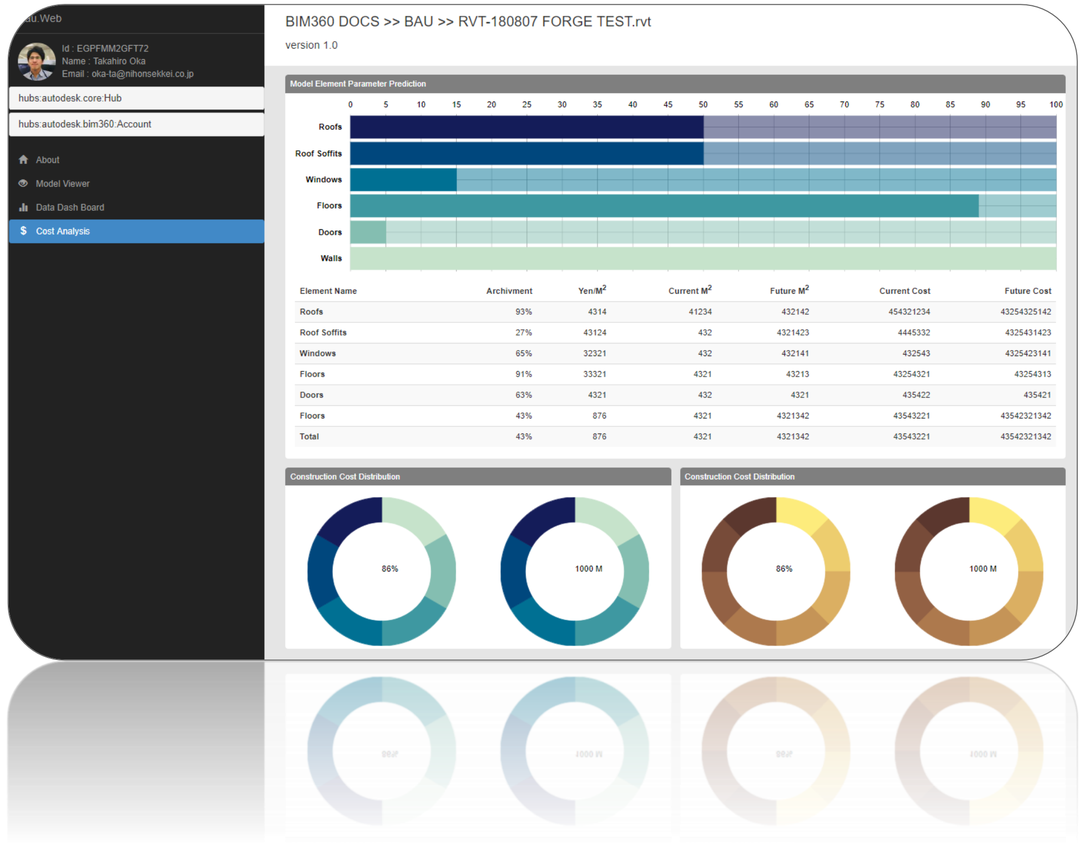
System where you want use model

```
C:¥Users¥oka-ta>curl -X POST http://127.0.0.1:5001/cost-rf  -d "total_floor_area=20000"
{
  "cost_total_prediction": [
    [
      4445709651.612666
    ]
  ],
  "regr_coef": [
    [
      244286.99180634372
    ]
  ],
  "regr_intercept": [
    -440030184.51420784
  ]
}
```

Export and Visualize Result



Use Model at Your System



About Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

Jupyter Notebook
(Source : http://jupyter.org/)

reserach

- Make reggression model

In [50]:

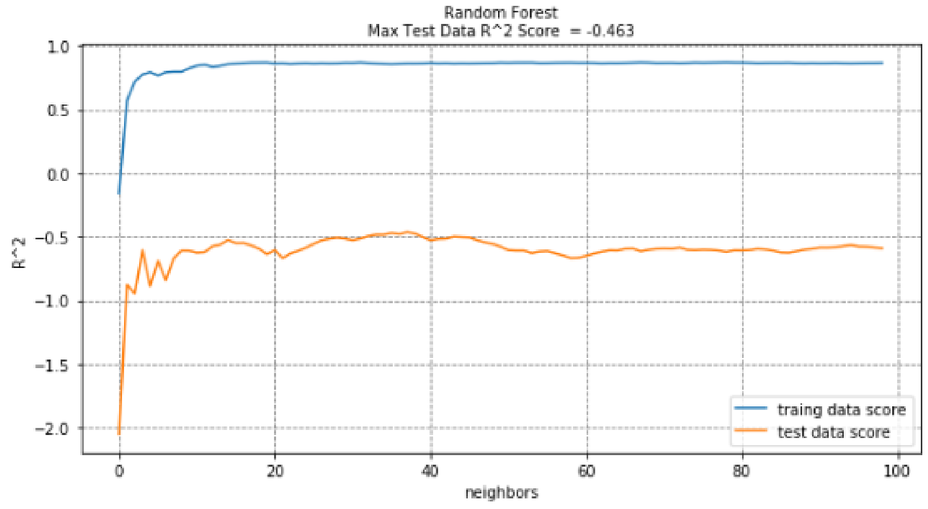
```
wall = pd.concat([cost_walls["cost_parameter_walls"],
                  cost_walls["cost_unitprice_walls"],
                  pd.DataFrame(project_total, columns=["sccs_total"])], axis=1).replace(0, np.nan).dropna(h
input_wall = pd.concat([wall["cost_parameter_walls"], wall["sccs_total"]], axis=1)
output_wall = wall["cost_unitprice_walls"]

X_wall_train, X_wall_test, y_wall_train, y_wall_test = train_test_split(input_wall, output_wall, random_sta
```

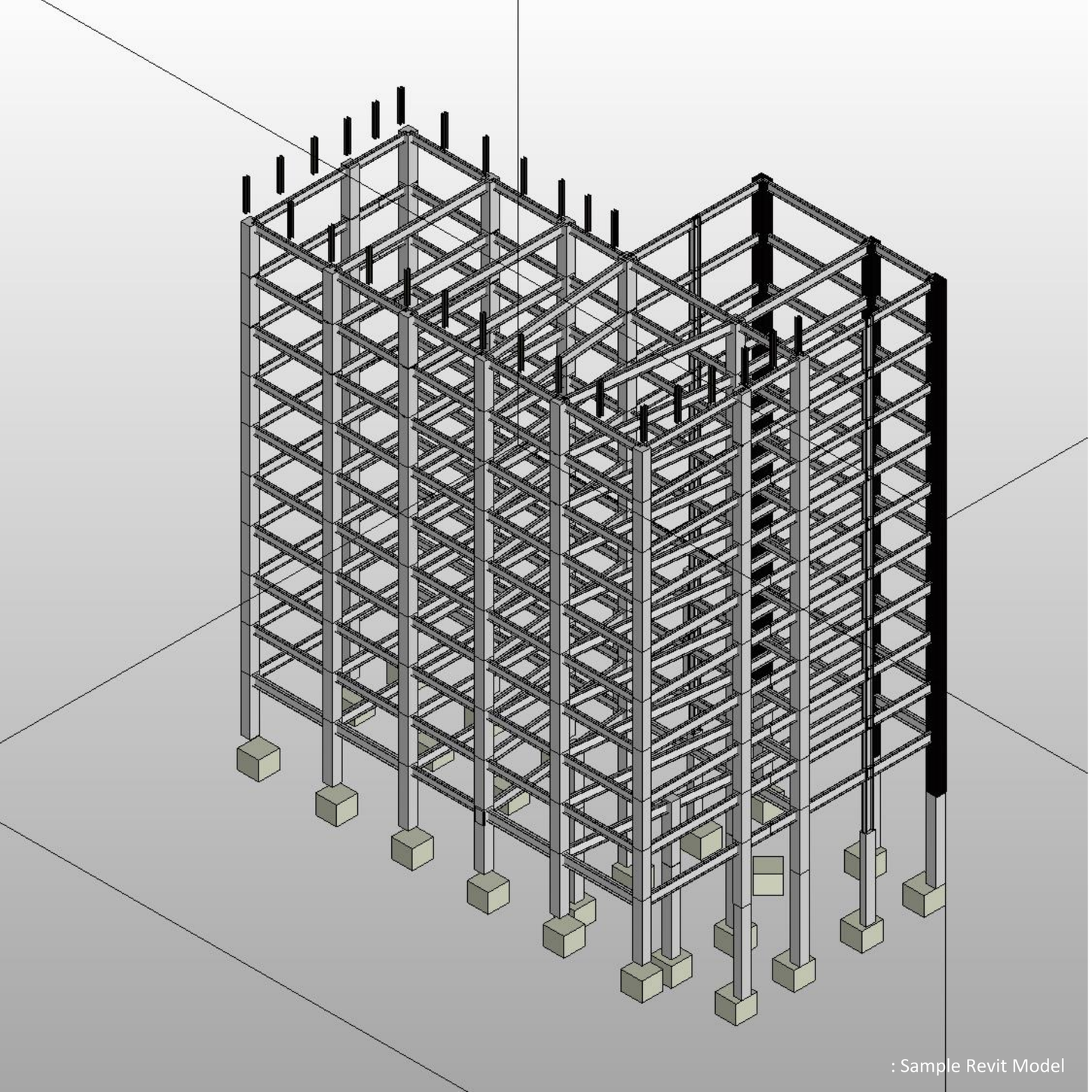
RandomForest

In [61]:

```
neighbors       = range(1,100)
r2_train_score  = []
r2_test_score   = []

for i in neighbors:
    regr = RandomForestRegressor(random_state=27, n_estimators=i)
    regr.fit(np.array(X_wall_train), np.array(y_wall_train))
    r2_train_score.append(regr.score(X_wall_train, y_wall_train))
    r2_test_score.append(regr.score(X_wall_test,  y_wall_test))

f,ax = plt.subplots(figsize=(10, 5))
ax.grid(which='major',color='gray',linestyle='--')
ax.plot(r2_train_score)
ax.plot(r2_test_score)
ax.set_title("Random Forest¥n Max Test Data R^2 Score  = {:.3f} ".format(max(r2_test_score)), size=10)
ax.set_ylabel("R^2")
ax.set_xlabel("neighbors")
ax.legend(["traing data score", "test data score"])
```

Out[61]: <matplotlib.legend.Legend at 0x2a5ef5f9c50>

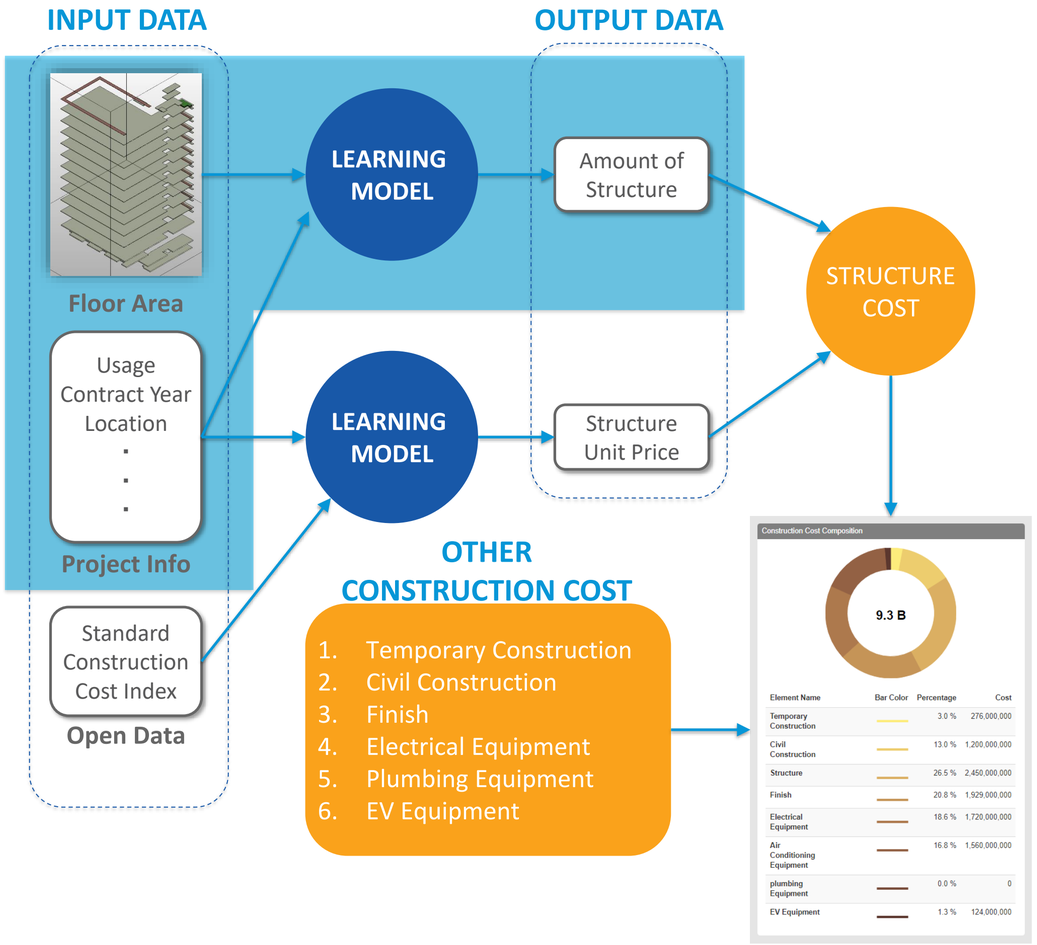


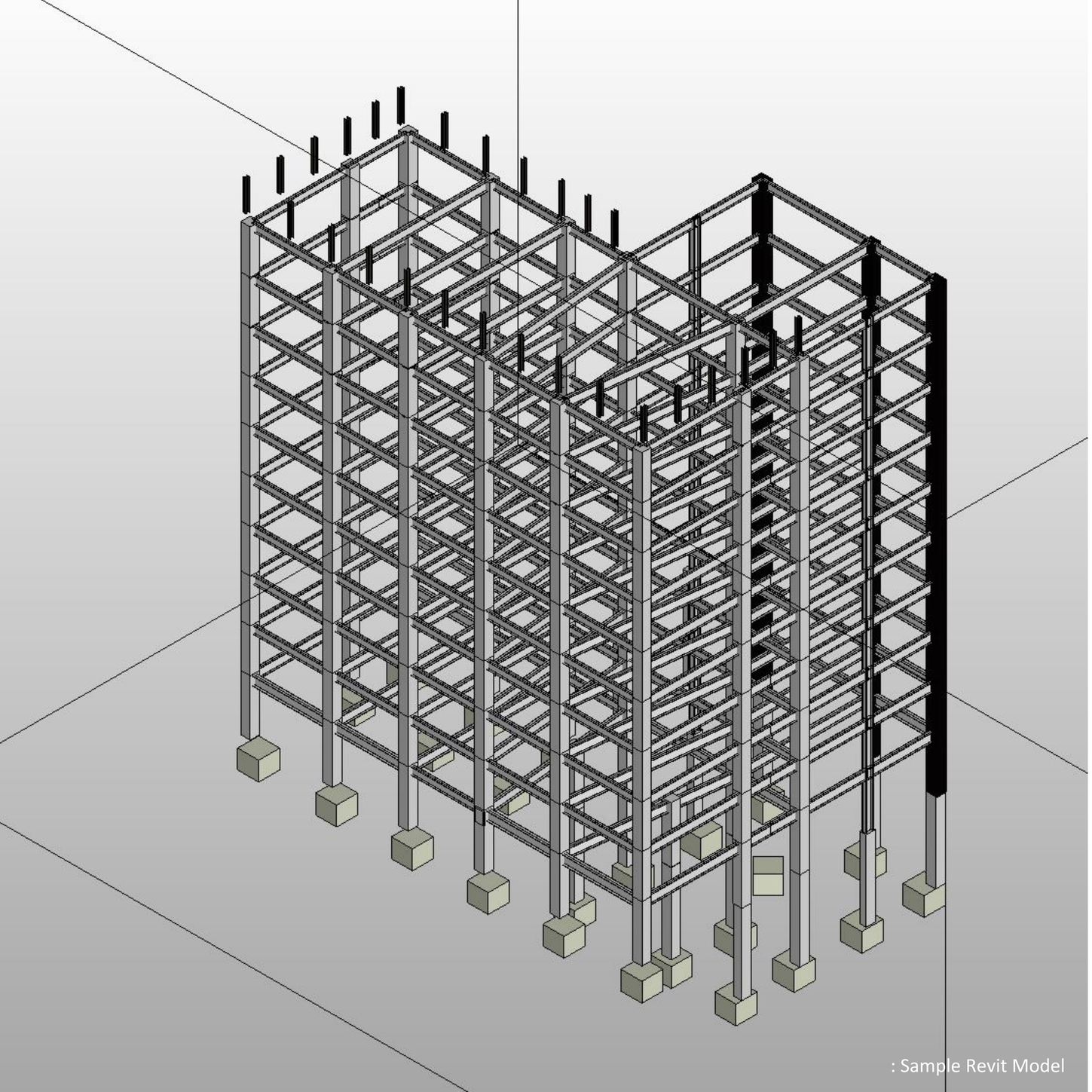


: Sample Revit Model

Example - 1

Construction Cost Estimation

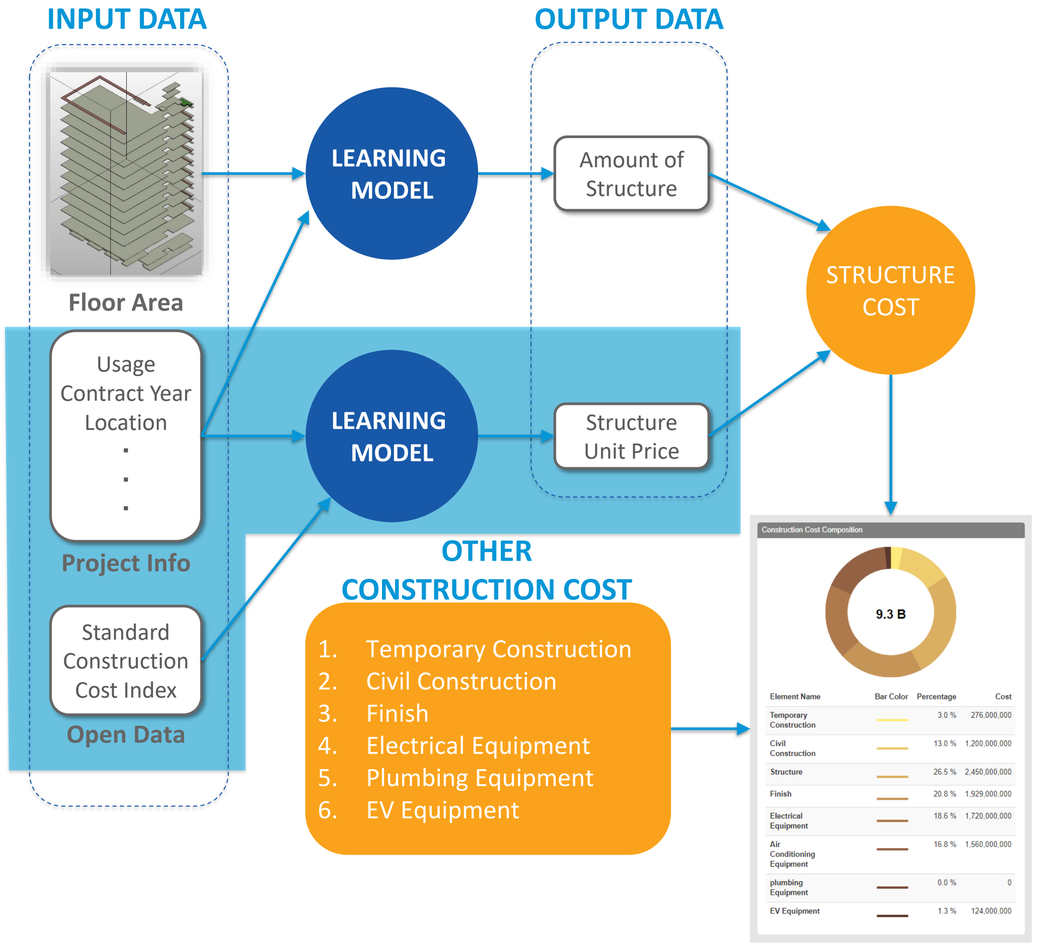




: Sample Revit Model

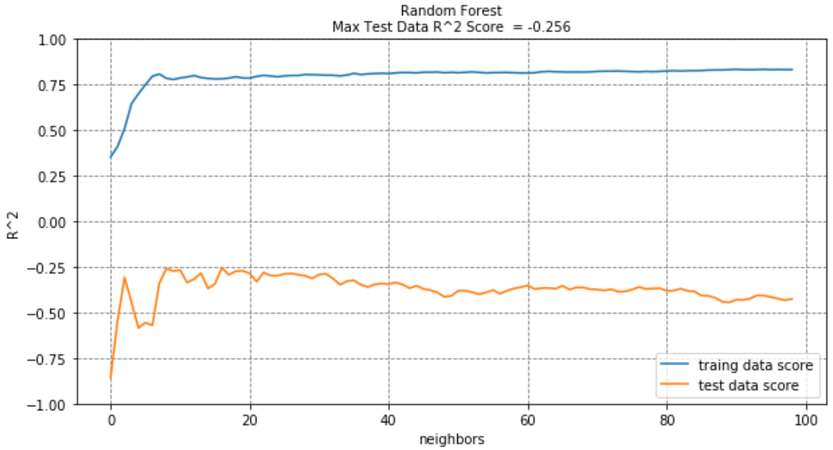
Example - 2

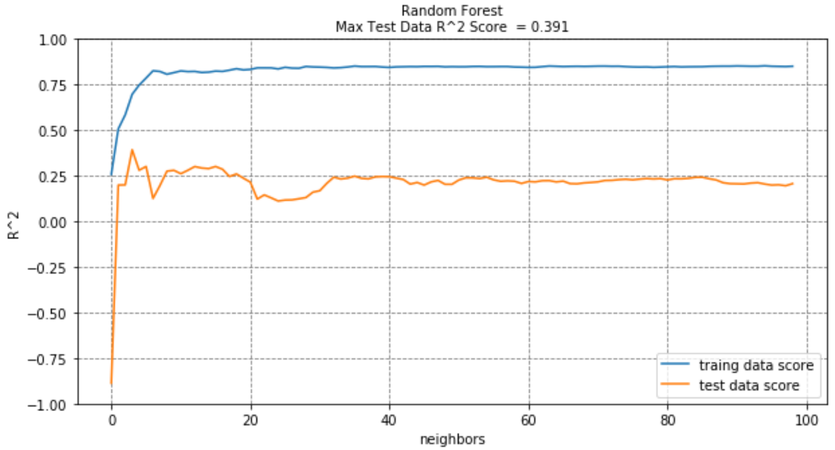
Construction Cost Estimation

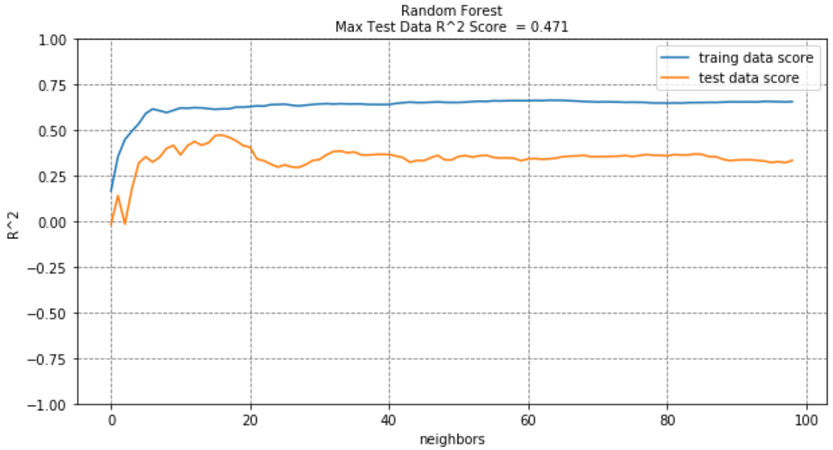


Jupyter Notebook Slide

How improve Accuracy?





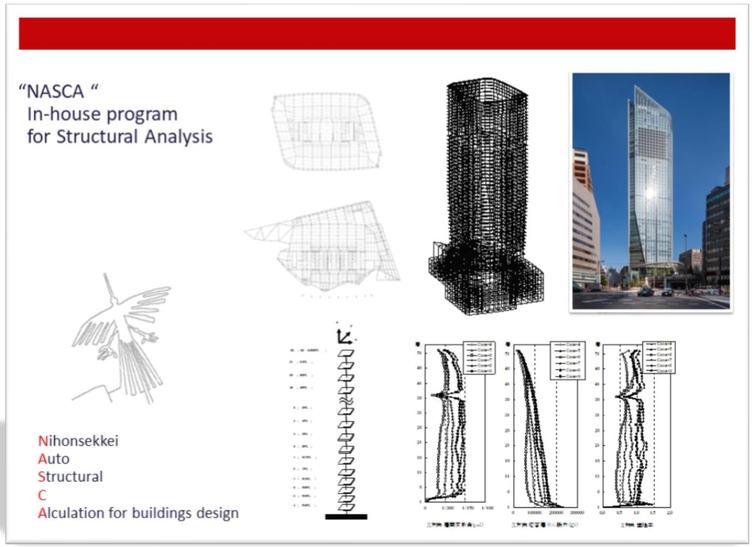


DATA ADDITION

ALGORITHM
PARAMETER CHINING

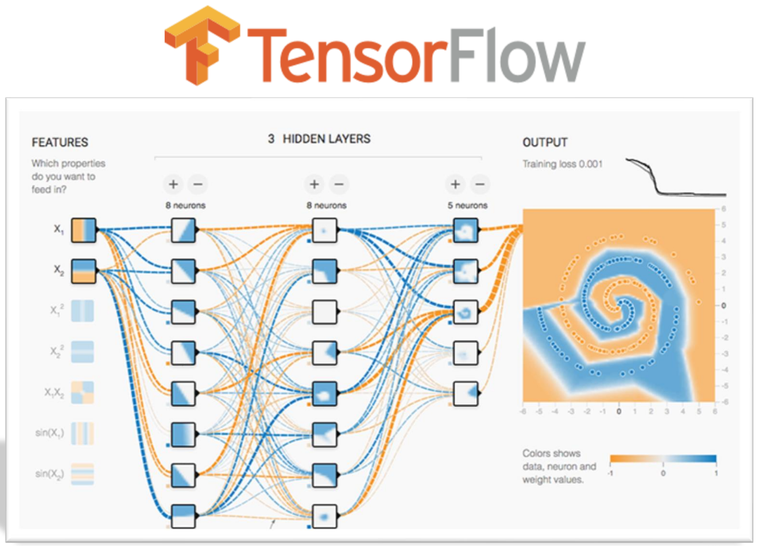
Future Outlook

Future Outlook



Data Analysis with Structure and MEP Engineering

NIHONSEKEEI, INC. have department of Structure and MEP, so we can extend this system for them. We will continue researching how use this system in their field such as predicting something spec of them for supporting decision-making.



Adopt Latest ML Algorithm called Deep Learning

We continue researching data analysis method for architectural field with not only legacy ML algorithm such as SVM and K-Neighbour but also latest ML which called Deep Learning that supported by Tensorflow and Keras.

Image Source :
https://becominghuman.ai/creating-your-own-neural-network-using-tensorflow-fa8ca7c

Question?

oka-ta@nihonsekkei.co.jp

AUTODESK®

Make anything™