Data platform for Smart Building using BIM and WoT

2021/10/11

Takashi Kasuya
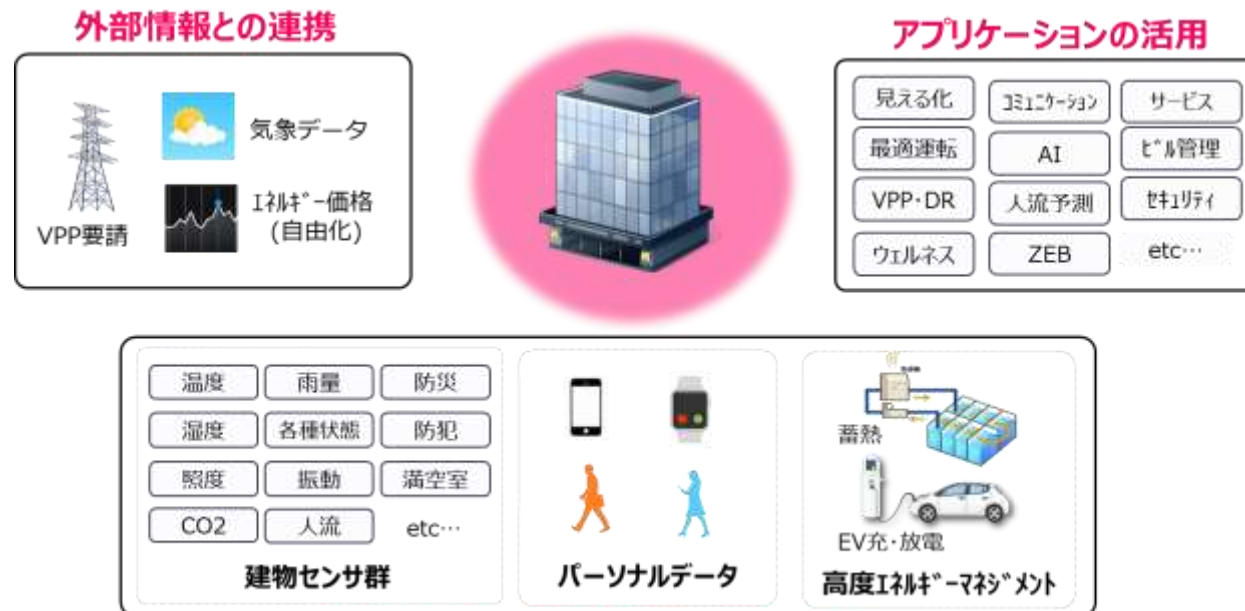ex. Tokyo University Esaki Lab.
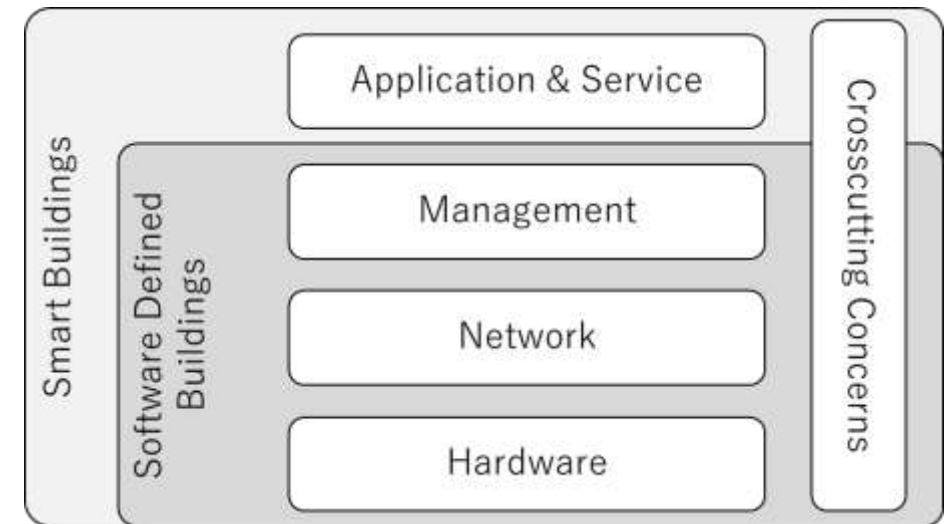Takenaka Corporation

TAKENAKA

竹中工務店

Smart Building

Buildings that achieve more advanced energy-saving performance and improved comfort through IoT and service coordination

functions

① Safety and Security Functions
② Energy-Efficiency Functions
③ Comfort Functions
  [Ergonomics of the Building]

④ Higher-Level Management Functions
⑤ Display and Operating Functions
  [User Interface]

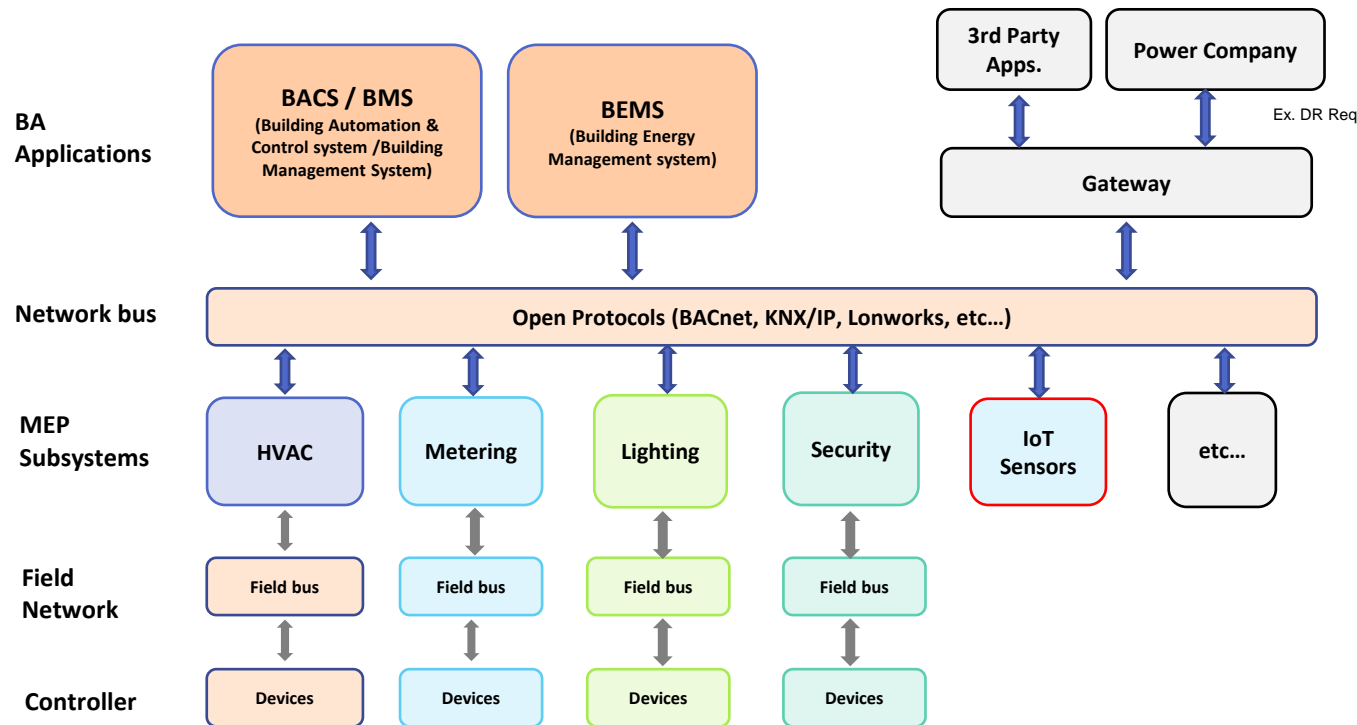(Bali, Smart Building Design: Conception, Planning, Realization, and Operation, 2019)

外部情報との連携

VPP要請　気象データ　エネルギー価格（自由化）

アプリケーションの活用

見える化　コミュニケーション　サービス
最適運転　AI　ビル管理
VPP・DR　人流予測　セキュリティ
ウェルネス　ZEB　etc…

温度　雨量　防災
湿度　各種状態　防犯
照度　振動　満空室
CO2　人流　etc…

建物センサ群　　パーソナルデータ

蓄熱
EV充・放電
高度エネルギーマネジメント

ビル内データ量の増加と高度な設備制御

Smart Buildings

Software Defined Buildings

Application & Service
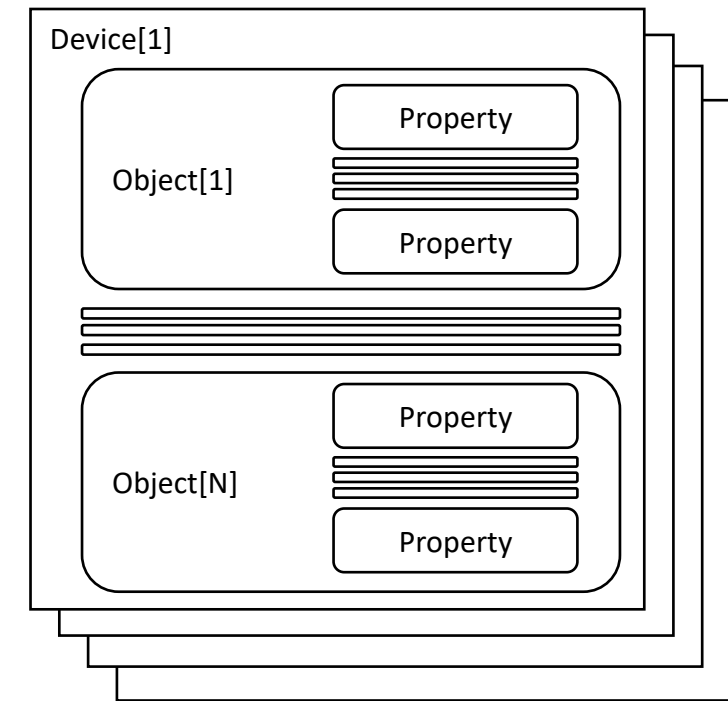
Management

Network

Hardware

Crosscutting Concerns

M.Mazzara, et.al,
Reference architecture for smart and software-defined buildings, 2019

BACS：Building Automation and Control System

① Multi-vendor systems, consists of HVAC, Metering and so on.

② Increasing number of cases with the Internet due to the demand for IoT and AI applications

③ BACnet has high barriers to entry for non-specialized vendors and low interoperability (Especially in Japan)

BA Applications	**BACS / BMS** (Building Automation & Control system /Building Management System) **BEMS** (Building Energy Management system) **3rd Party Apps.** **Power Company** Ex. DR Req. **Gateway**
Network bus	Open Protocols (BACnet, KNX/IP, Lonworks, etc…)
MEP Subsystems	HVAC Metering Lighting Security IoT Sensors etc…
Field Network	Field bus Field bus Field bus Field bus
Controller	Devices Devices Devices Devices

建物設備の構成イメージ

Device[1]

Object[1] — Property / Property

Object[N] — Property / Property
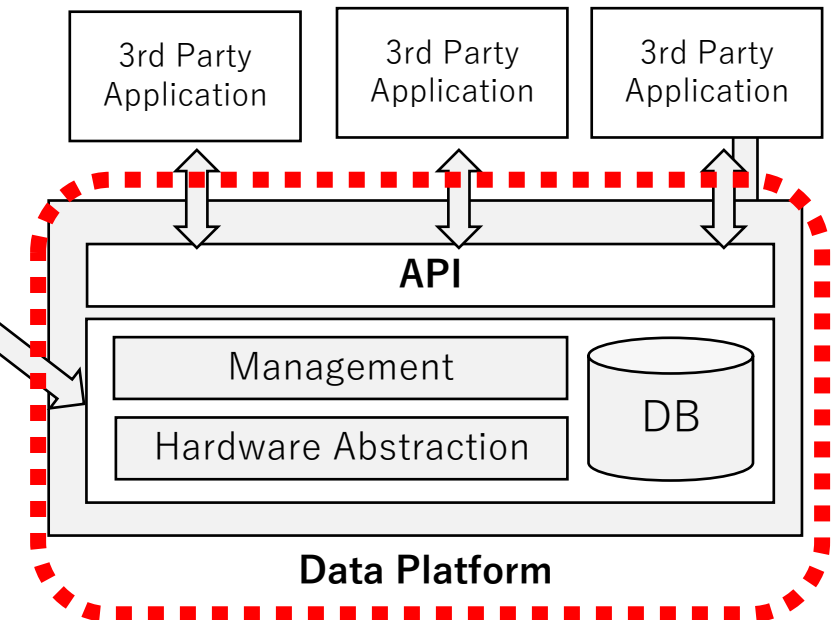
BACnetのデータモデル

TAKENAKA

Software Defined BACS

- **While maintaining the functionality of BACS, which has a data platform with hardware abstraction and data management capabilities to exceed its functional limits.**
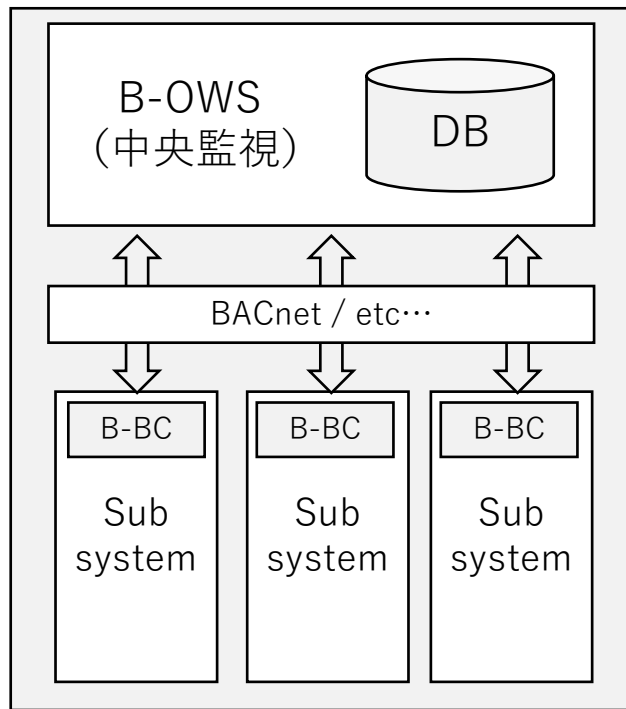

Traditional BACS
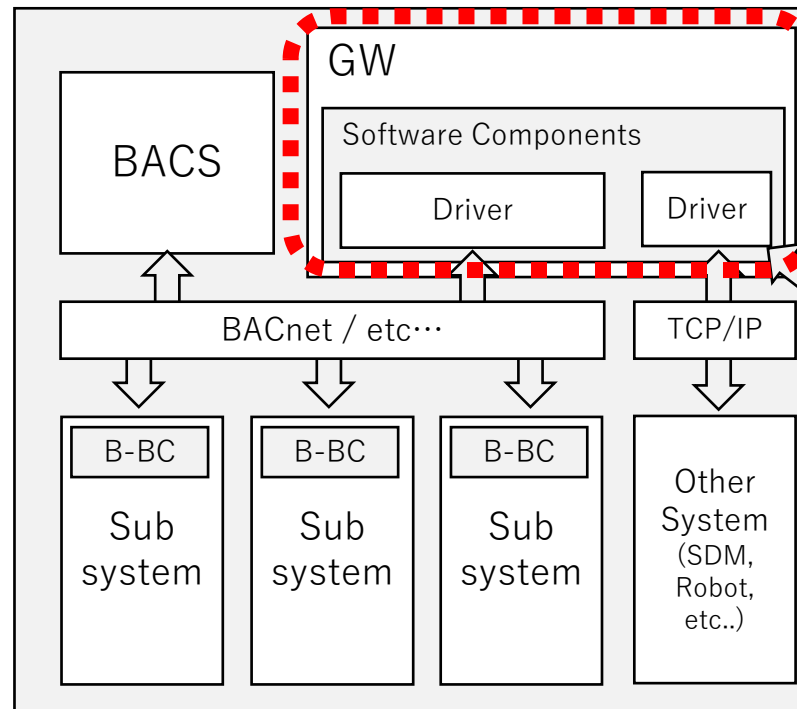
Software Defined BACS

Based on many projects, define the optimal API (Application Programming Interface) for smart buildings in consideration of industry standards

BIM（Building Information Modeling）

- A digital representation of the physical and functional characteristics of a facility, shared as a knowledge resource of facility information, and forming the basis of confidence for decision making throughout the life cycle of a facility

- BACS vendors rarely use BIM.
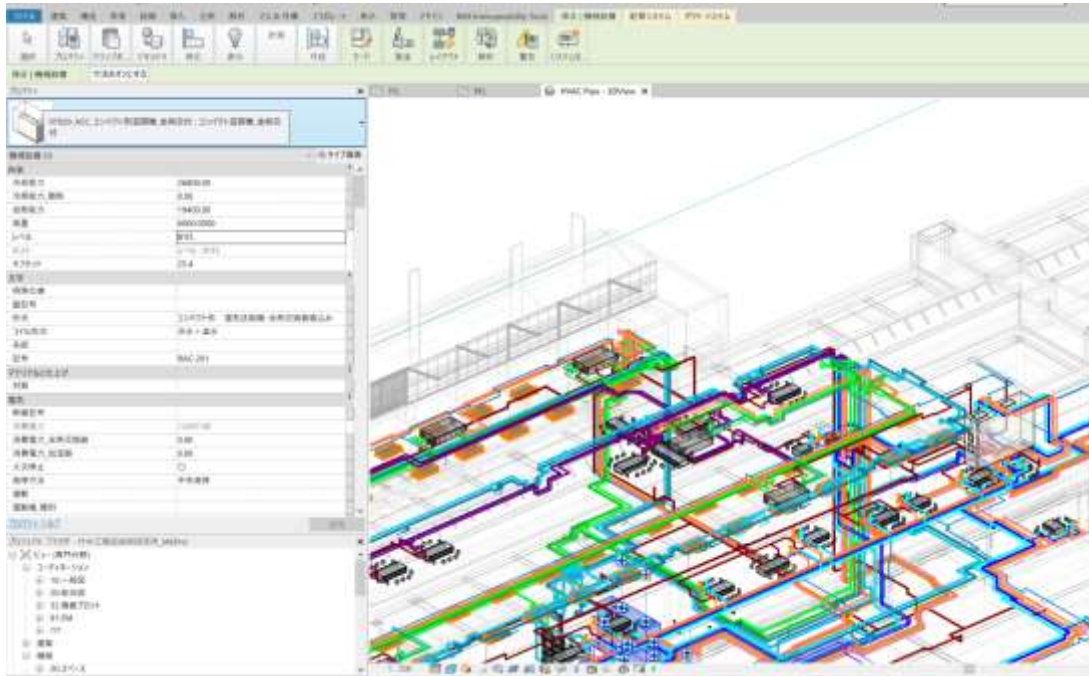
Authoring tool for BIM（Autodesk Revit）
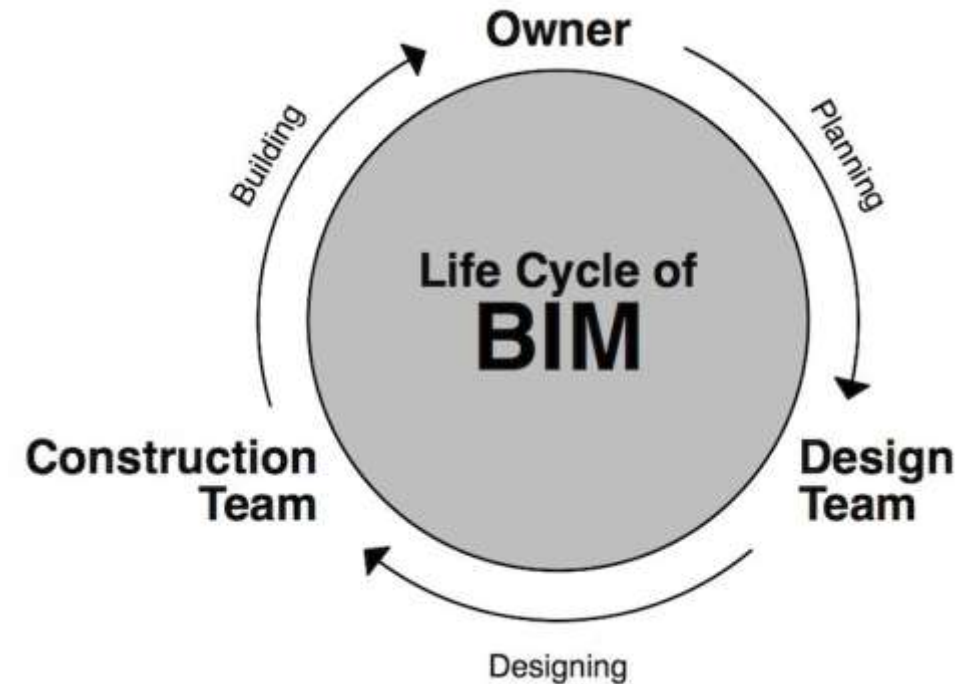
Lifecycle for BIM
http://www.shoegnome.com/2014/10/30/openbim/

Issues for Smart Building Data Platform

① Data Modeling
   i.   BIM and BACS point list

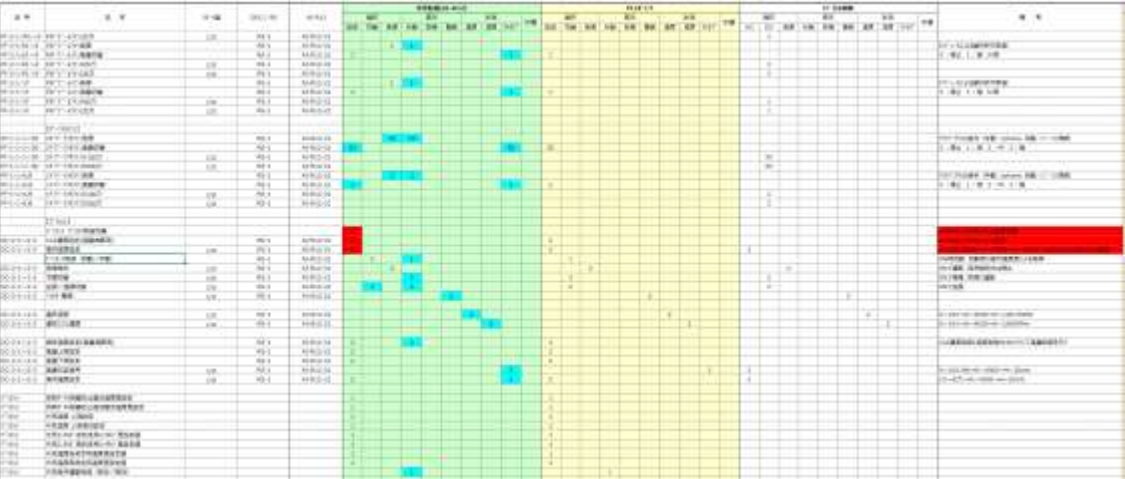## ② Support for practical use cases that consider IoT and AI
   i.   Lambda Architecture
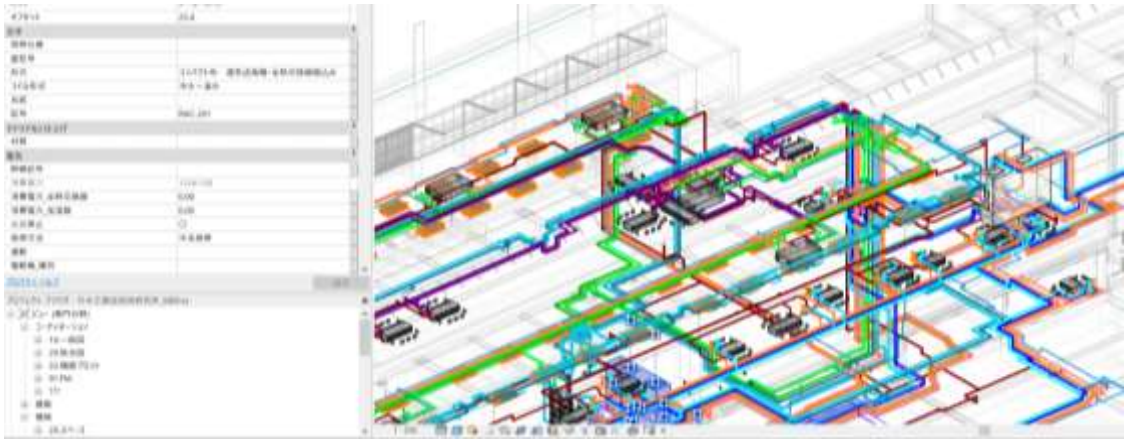   ii.   Web of things

③ Reduction of running costs
   i.   HDFS (Hadoop Distributed File System)
   ii.   PaaS (Platform as a Service) native
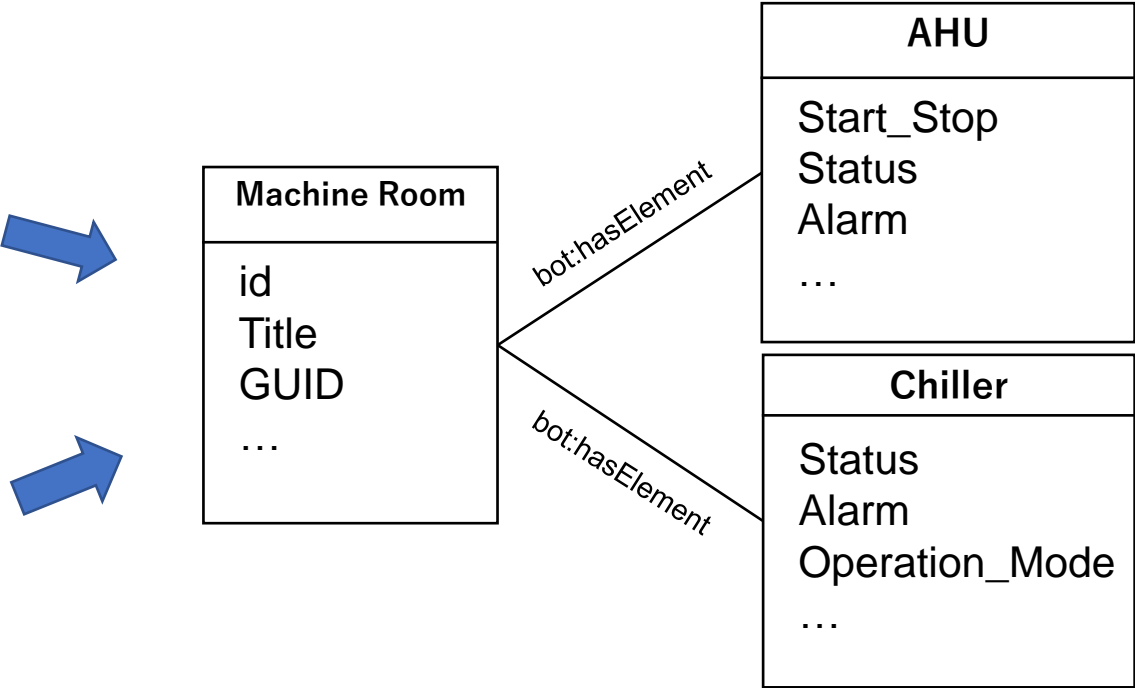
Proposed Method：Data modeling automation

Obtain spatial hierarchies from BIM and give relationships between spaces and devices

BACS point list（defined in ISO16484-3）

BIM (IFC)

Machine Room

id
Title
GUID
…

bot:hasElement

bot:hasElement

AHU

Start_Stop
Status
Alarm
…

Chiller

Status
Alarm
Operation_Mode
…

Extract Geometry / Metadata from BIM(IFC)

IFCのすべてを抽出することは不要なので、IFCのtypeを
参照しながら、適宜フィルタリングを実施

```
#2858= IFCSPACE('3q4P0iC9P33P2WOFzvBOXb',#41,'51',$,$,#2826,#2856,'\X2\30C830A430EC\X0\',.ELEMENT.,.INTERNAL.,$);
#2861= IFCPROPERTYSINGLEVALUE('Reference',$,IFCIDENTIFIER('\X2\30C830A430EC\X0\ 51'),$);
#2862= IFCPROPERTYSET('2FNT1K8tLDogItFDZWyAQR',#41,'Pset_SpaceCommon',$,(#2861));
#2864= IFCRELDEFINESBYPROPERTIES('24w_Q7l$9AwOktuDm126SJ',#41,$,$,(#2858),#2862);
#2868= IFCAXIS2PLACEMENT3D(#6,$,$);
#2869= IFCLOCALPLACEMENT(#157,#2868);
#2870= IFCCARTESIANPOINT((540.,125.));
#2872= IFCCARTESIANPOINT((990.,125.));
#2874= IFCCARTESIANPOINT((990.,290.));
#2876= IFCCARTESIANPOINT((-1530.,290.));
#2878= IFCCARTESIANPOINT((-1530.,-415.));
#2880= IFCCARTESIANPOINT((540.,-415.));
#2882= IFCPOLYLINE((#2870,#2872,#2874,#2876,#2878,#2880,#2870));
#2884= IFCARBITRARYCLOSEDPROFILEDEF(.AREA.,$,#2882);
#2885= IFCCARTESIANPOINT((15850.57102928,2272.03669197011,0.));
#2887= IFCAXIS2PLACEMENT3D(#2885,$,$);
#2888= IFCEXTRUDEDAREASOLID(#2884,#2887,#19,3500.);
#2889= IFCSHAPEREPRESENTATION(#102,'Body','SweptSolid',(#2888));
#2891= IFCPRODUCTDEFINITIONSHAPE($,$,(#2889));
#2893= IFCSPACE('3q4P0iC9P33P2WOFzvBOXa',#41,'52',$,$,#2869,#2891,'\X2\90E85C4B\X0\',
#2896= IFCPROPERTYSINGLEVALUE('Reference',$,IFCIDENTIFIER('\X2\90E85C4B\X0\ 52'),$);
#2897= IFCPROPERTYSET('3mdLZOPPH8X8BQi8MpoMd8',#41,'Pset_SpaceCommon',$,(#2896));
#2899= IFCRELDEFINESBYPROPERTIES('1gD3lnfW95EAp56XusjpCv',#41,$,$,(#2893),#2897);
#2903= IFCAXIS2PLACEMENT3D(#6,$,$);
```

IFC(IFC2X3)データ

```
{
 "@id": "http://hongo.wide.ad.jp/001/51_2858",
 "@type": "https://w3id.org/bot#Space",
 "GlobalId": "3q4P0iC9P33P2WOFzvBOXb",
 "InteriorOrExteriorSpace": "INTERNAL",
 "LongName": "トイレ",
 "Name": "51",
 "containsElement": [
   {"@id": "http://hongo.wide.ad.jp/001/小便器_-_壁掛け_185515"},
   {"@id": "http://hongo.wide.ad.jp/001/カウンタートップ_135950"},
   {"@id": "http://hongo.wide.ad.jp/001/洋式便器_6_135895"},
   {"@id": "http://hongo.wide.ad.jp/001/洗面器_1_135851"},
   {"@id": "http://hongo.wide.ad.jp/001/洗濯流し_2_135873"},
   {"@id": "http://hongo.wide.ad.jp/001/洋式便器_6_135827"},
   {"@id": "http://hongo.wide.ad.jp/001/小便器_-_壁掛け_185537"}
 ],
 "id": 2858,
 "type": "IfcSpace"
}
```
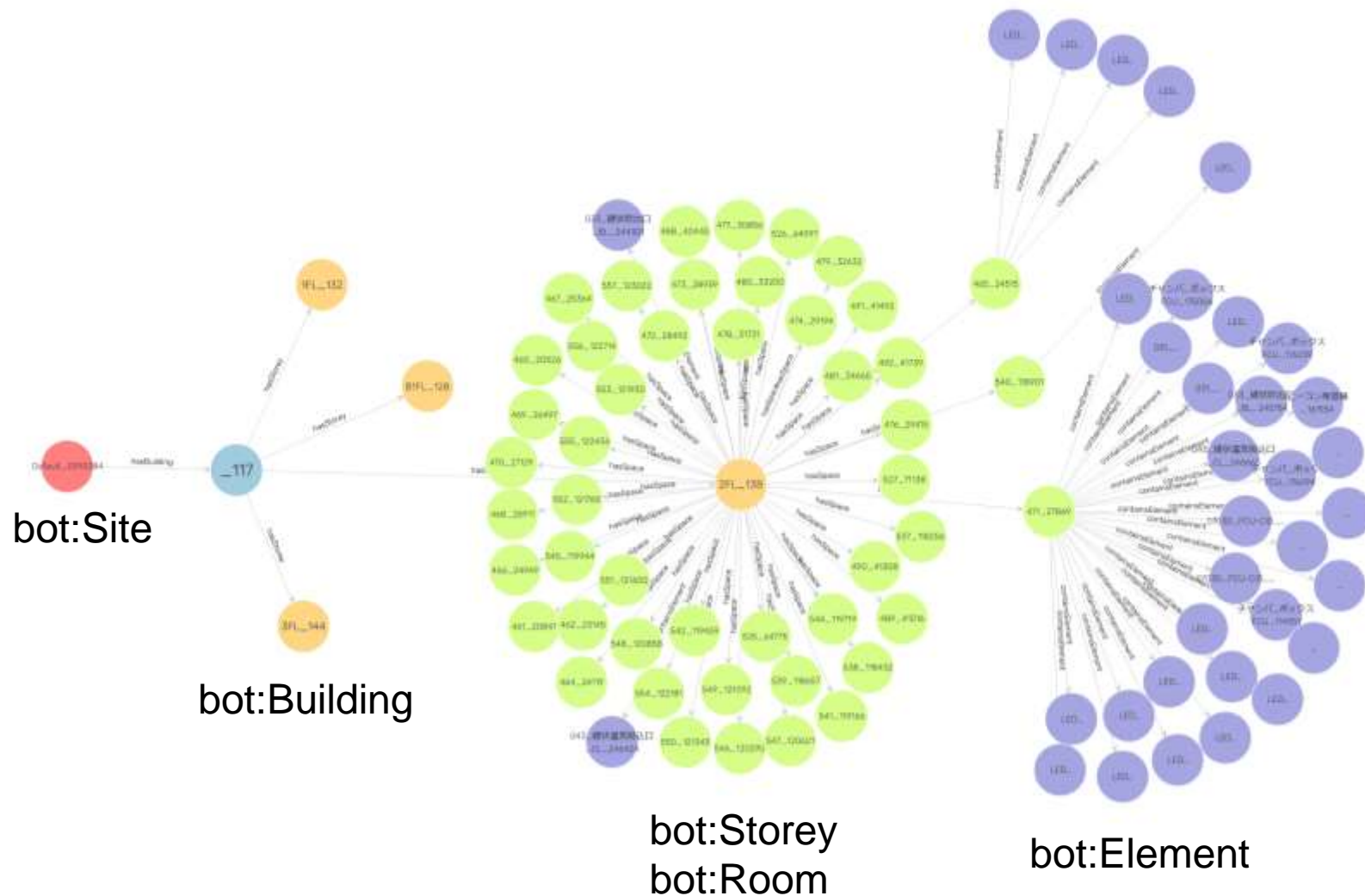
RDF(JSON-LD)

空間グラフを作成し、IFCから抽
出した属性データや形状データ
への参照（GUID）も付与する。
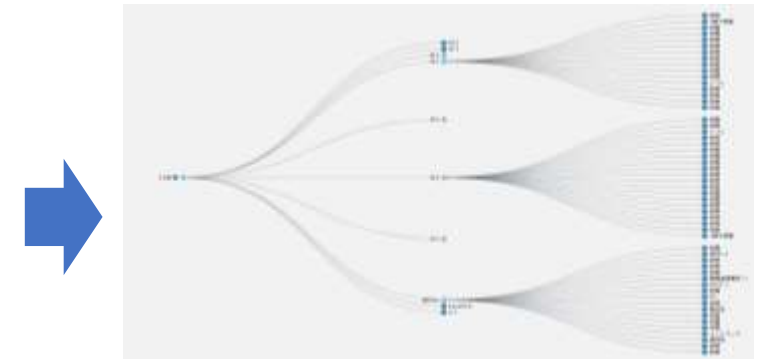
Space Graph/metadata
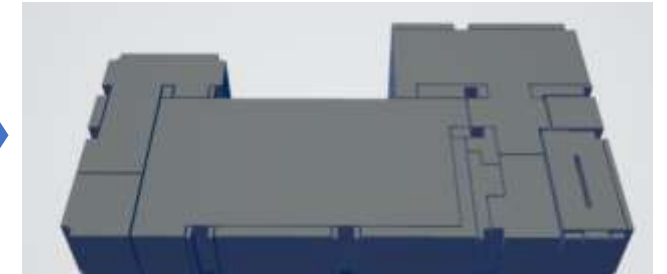
Geometries

Visual graph ⓘ

bot:Site

bot:Building

bot:Storey
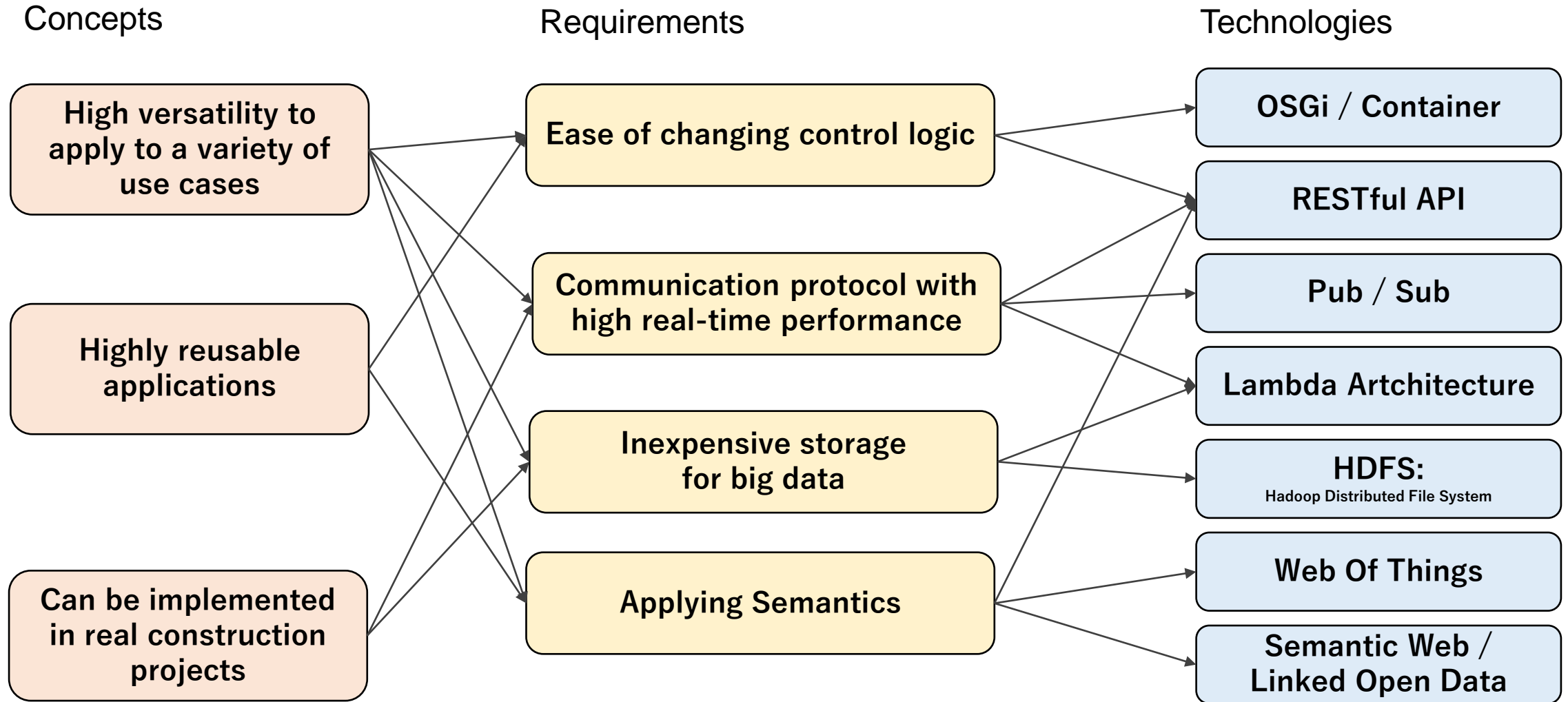bot:Room
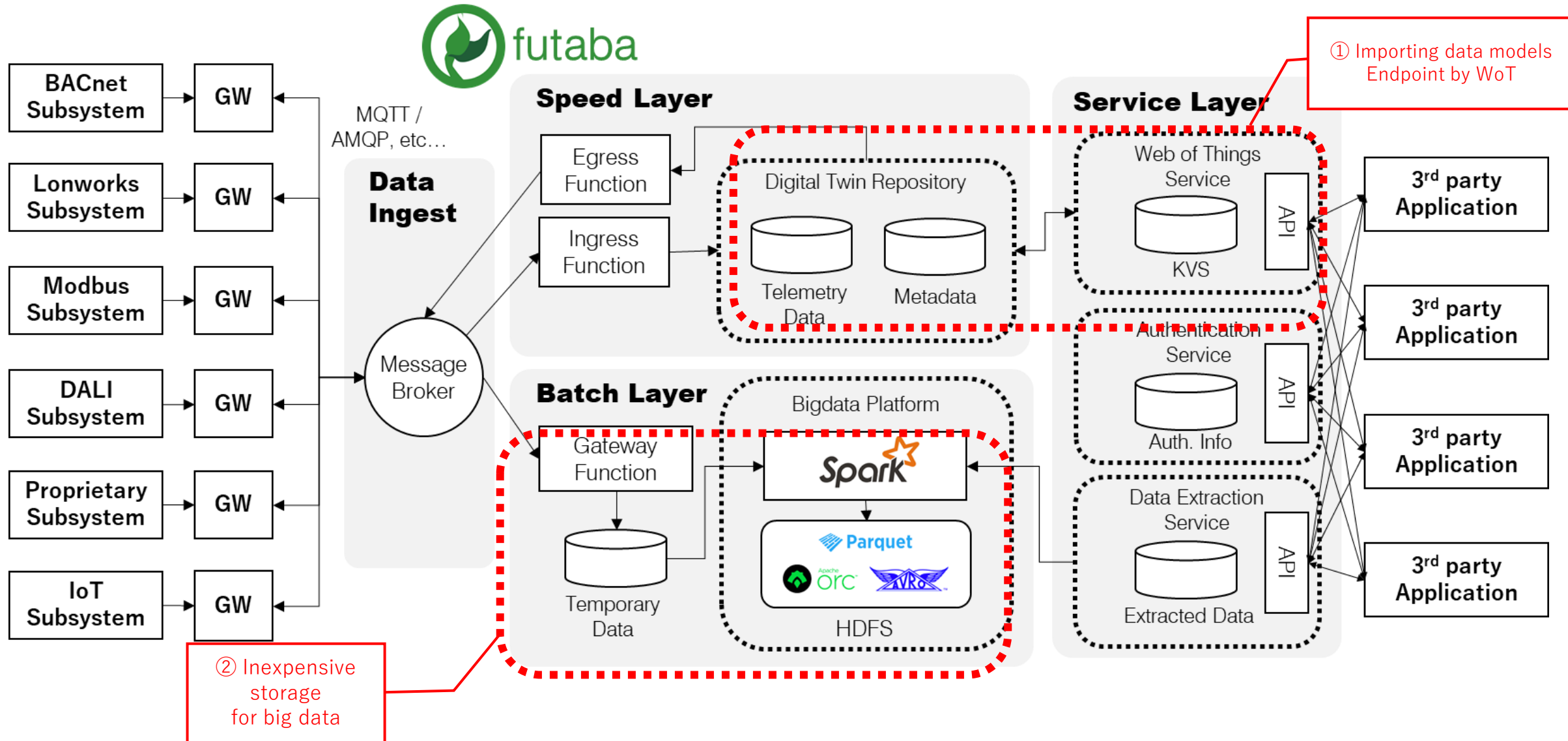
bot:Element

ビーコン発信機_161934

🏷 ビーコン発信機_161934

Types:
bot:Element

RDF rank:
0

🔍 Search instance properties

http://www.buildingsmart-tech.org/IFC2x3#GlobalId
0ye0603fb9iefo3foPjmed

http://www.buildingsmart-tech.org/IFC2x3#Name
ビーコン発信機:電池型:17080707

http://www.buildingsmart-tech.org/IFC2x3#ObjectType
電池型

http://www.buildingsmart-tech.org/IFC2x3#Tag
17080707

http://www.buildingsmart-tech.org/IFC2x3#id
161934

http://www.buildingsmart-tech.org/IFC2x3#type
IfcBuildingElementProxy

http://www.buildingsmart-tech.org/IFC2x3#オフセット
1.1e+03

http://www.buildingsmart-tech.org/IFC2x3#カテゴリ
通信機器

http://www.buildingsmart-tech.org/IFC2x3#タイプ
ビーコン発信機: 電池型

http://www.buildingsmart-tech.org/IFC2x3#タイプID
ビーコン発信機: 電池型

http://www.buildingsmart-tech.org/IFC2x3#ビーコンNo
5M

Requirement / Design for Smart Building Platform

Concepts

Requirements

Technologies

Concepts	Requirements	Technologies
High versatility to apply to a variety of use cases	Ease of changing control logic	OSGi / Container
Highly reusable applications	Communication protocol with high real-time performance	RESTful API
Can be implemented in real construction projects	Inexpensive storage for big data	Pub / Sub
	Applying Semantics	Lambda Artchitecture
		HDFS: Hadoop Distributed File System
		Web Of Things
		Semantic Web / Linked Open Data

BIM を用いた Software Defined BACS の実現に関する研究

Architecture（**futaba** : Facility Unified digital-Twin Architecture for Building Automation）

Comparison of standards

Standardization	Standard Specification Range		
	Architecture	Transport	Data model
IEEE1888	規定あり	SOAP / HTTP	XML
oneM2M	TS-001	TS-0004	TS-0023 (Home Appliances)
ITU-T SG20	Y.4409/Y.2070	-	-
IIC	IIRA	Connectivity Framework (Only concept)	
W3C	WoT Architecture	WoT Protocol Binding	WoT Things Description
Echonet Consortium	(Out of scope)	ECHONET Lite	オブジェクト詳細規定
Device WebAPI Consortium		OMA GotAPU	OMA DWAPI-PCH
IPSO Alliance		OMA LWM2M	Smart Object
KNX Association		KNX	Application Description
BACnet		RS485/TCP	規定あり
BACnet / WS	規定あり	REST / HTTP	規定あり
Oracle IoT Cloud	規定あり	REST / HTTP	Oracle device model
futaba	**Lambda Architecture**	**MQTT**	**WoT Things Description**

Fujitsu WoT資料より一部追記

Data model / Endpoint using Web of Things

1. Point data is abstracted into a device-level data model and imported as a Thing.

2. Define inheritable class schemas with flexible telemetry formats, to support a variety of data types
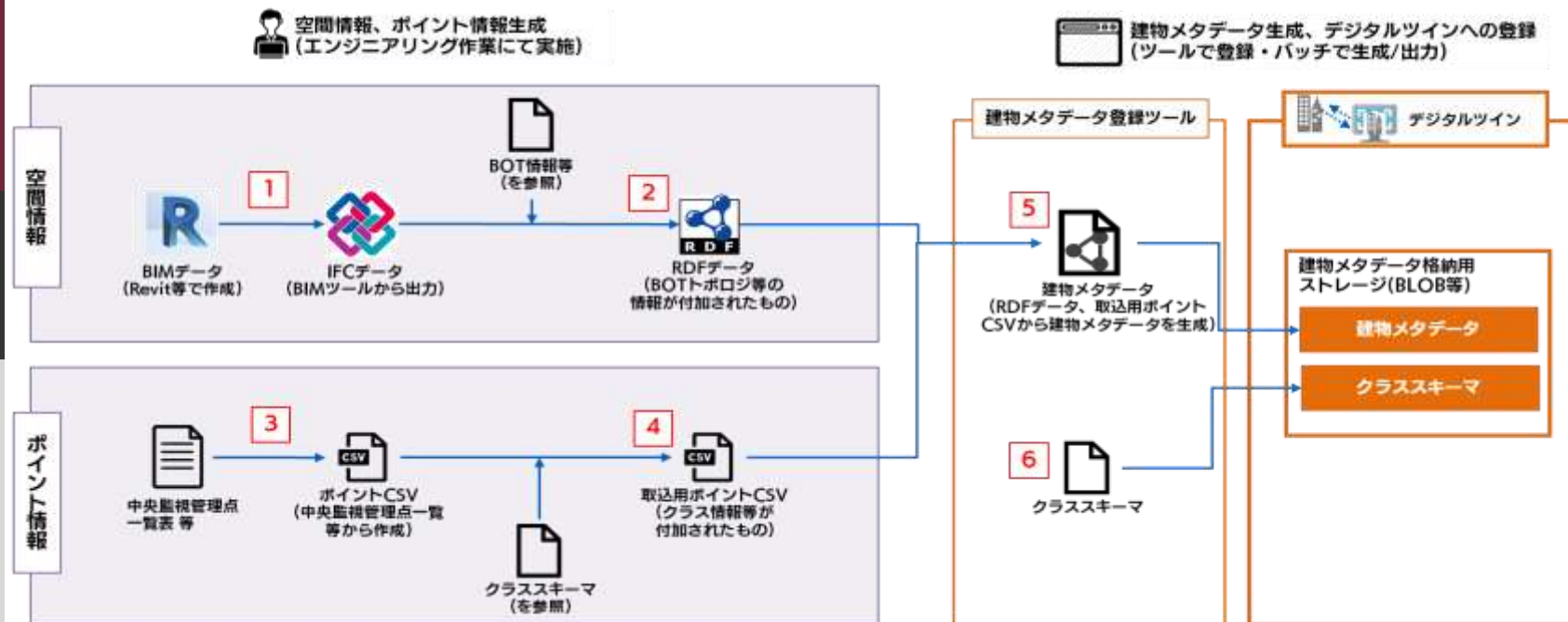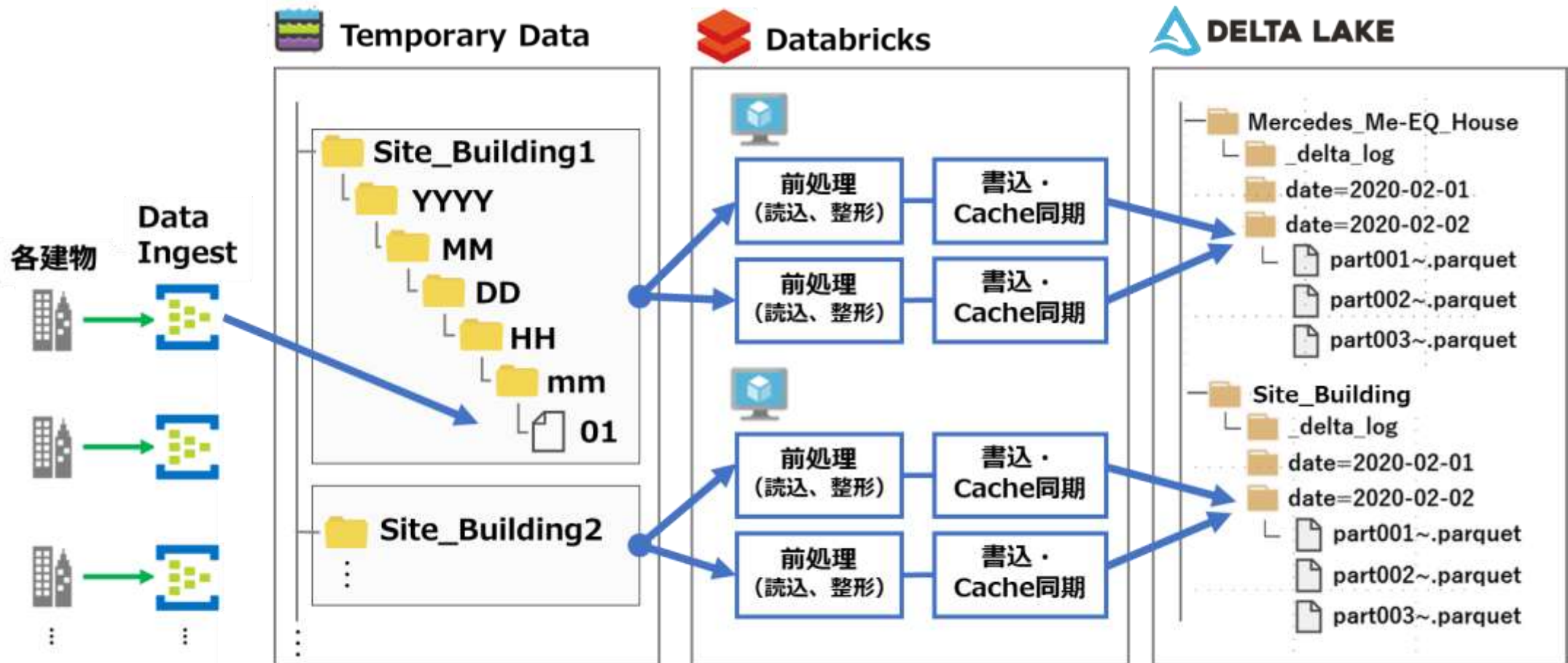

Generation Flow of Thing Description (TD) instance

Class Schemas for TD

Big data processing

Use case（1）：EQ House

演出システム

ウェアラブルデバイス
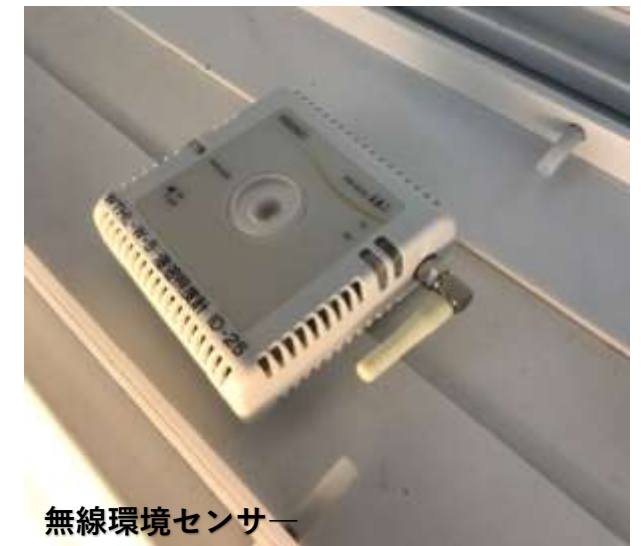
無線環境センサー

Use case ：EQ House (2)

Visual graph ⓘ
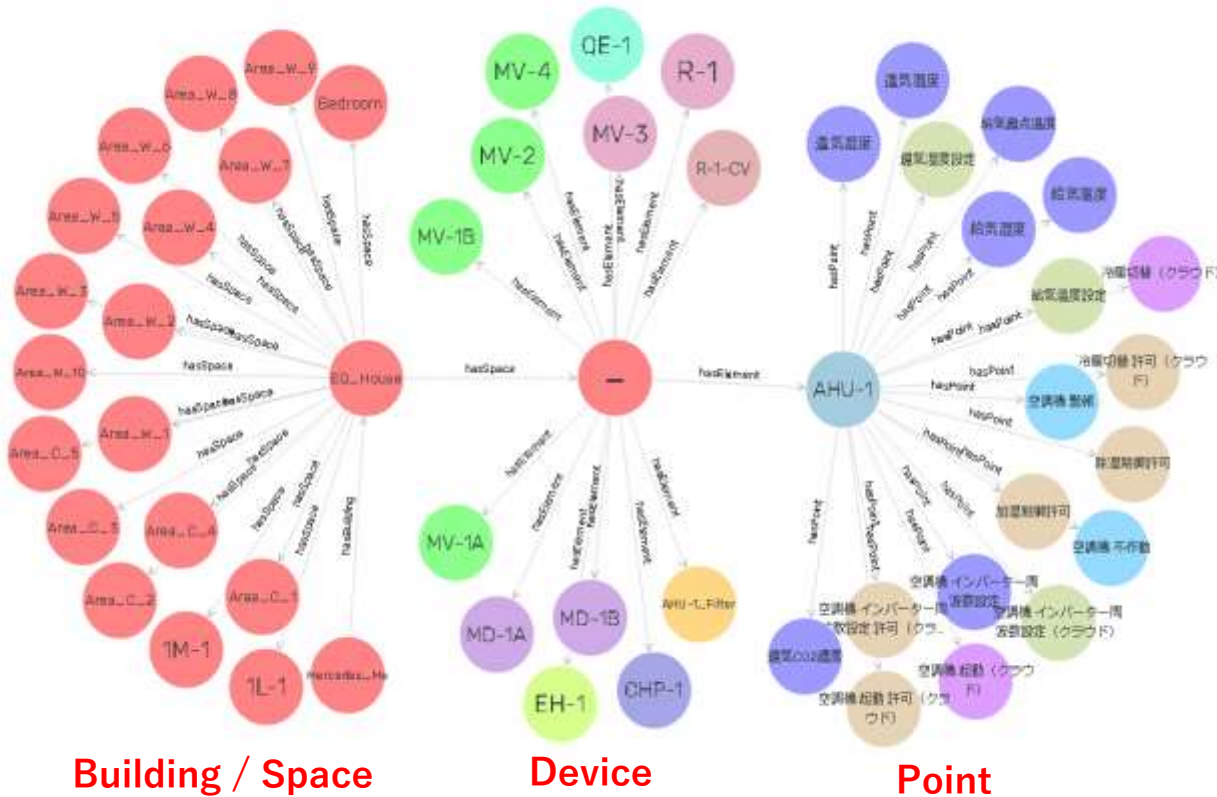
Building / Space   Device   Point

Space Graph / Data Model

Metadata / Things Description
(when accessed from a browser)

Use case：EQ House (3)

Remote control system using AI to improve comfort using PMV as an indicator.

Archiphilia Engine
1. imitation of human control using neural networks.
2. model updating and control by reinforcement learning
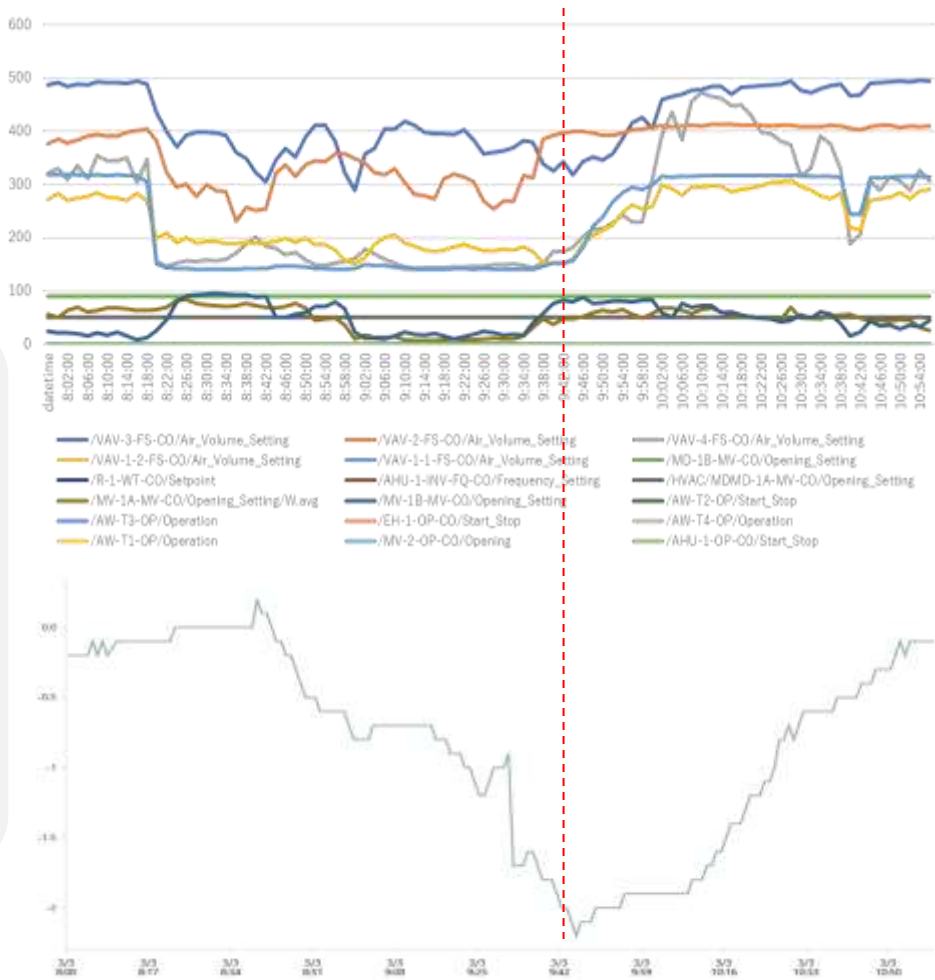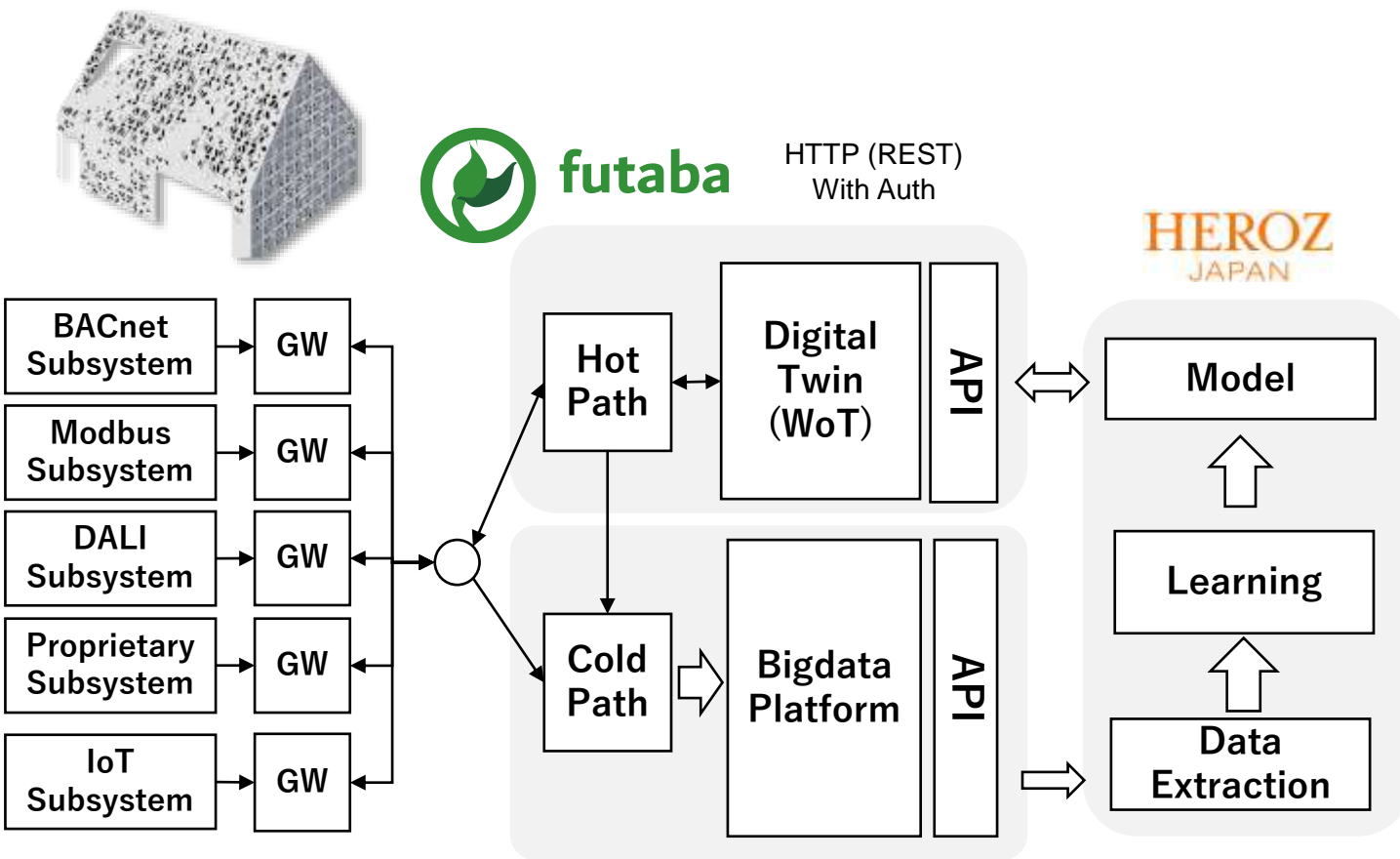
Results of remote control (top: control parameters, bottom: PMV)

Use case(2) : Takenaka Research & Development Institute

① AI remote control of lighting and HVAC (**27%** reduction in lighting, **14.5%** reduction in HVAC)

② Display sensor information on the digital twin app based on the BIM.

③ various other apps are being developed and tested
(robot collaboration, voice recognition, etc..)
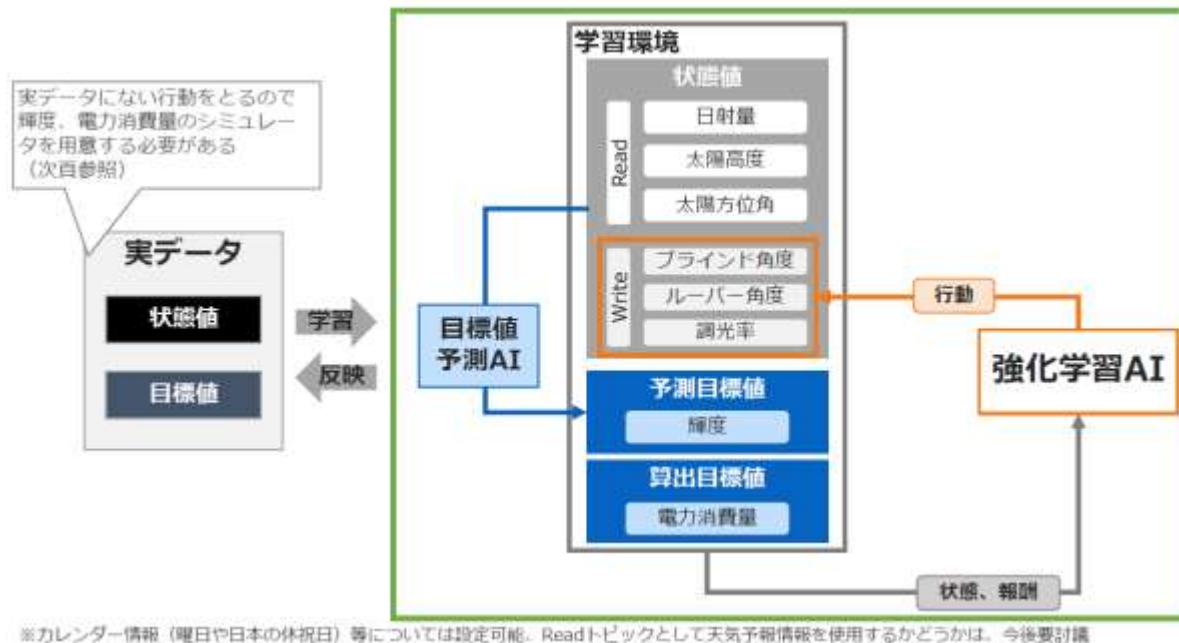
Reinforcement Learning Engine by HEROZ

Digital Twins Application

API

カテゴリ	機能	API	内容
HOT	WoT	TD取得	TD(Things Description)情報を取得
		Property取得	対象デバイスのプロパティ(ポイント)情報を取得
		Property書き込み	対象デバイスのプロパティ(ポイント)へ値をセット(書き込み)する
		Action 実行	メール送信等予め定義されたアクション(操作)を実行
		Eventサブスクライブ登録	API経由ではなく、メッセージングサービスからリアルタイムデータの受信を行う設定を登録
		Eventサブスクライブ状況取得	登録されたサブスクライブ情報を取得
		Eventサブスクライブ解除	登録済みサブスクライブ情報の解除(削除)
	WoT 拡張	Property拡張取得	対象プロパティの値情報だけではなく、プロパティ名情報も付加して取得
		Property一括取得	指定条件に合致する複数プロパティ情報を一括で取得
		Event一括サブスクライブ登録	指定条件に合致する複数プロパティ情報を一括でメッセージングサービスから取得する設定を登録
		Event一括サブスクライブ状況取得	登録されたサブスクライブ情報を取得
		Event一括サブスクライブ解除	登録済みサブスクライブ情報の解除(削除)
	建物メタデータ	建物メタデータ検索	条件を指定し、建物メタデータの検索と閲覧を行う
		建物メタデータ編集	建物メタデータ上の項目を編集(更新)する
COLD	モデル学習	タスク作成	AIエンジン等等向けにモデル学習用のデータ生成タスクを要求する
		タスクキャンセル	モデル学習データ生成タスクのキャンセルを行う
		タスク詳細確認	登録済みモデル学習データ生成タスクの実行状況、スケジュール状況を確認する
		タスク有効変更	登録済みモデル学習データの有効化/無効化を行う
		Webhook登録	モデル学習データ生成結果をWebHookで取得する際の通知先WebHookアドレスを登録する
		Webhook解除	登録済みWebHookアドレスの削除を行う
	共有データ	共有データ追加	3rd Partyシステム間でAPIを介したデータの受渡を行う際のデータ登録を行う
		共有データ検索	登録されたデータの検索を行う
		共有データ削除	登録されたデータの削除を行う

WoT API (1) Authentication

Create ID / Secret

↓

Request to Issue / Refresh
an access token

↓

Search Metadata and find TDs
to be monitor or controlled

↓

Obtain the target TDID from the
metadata. GET or PUT properties
by using them as keys.
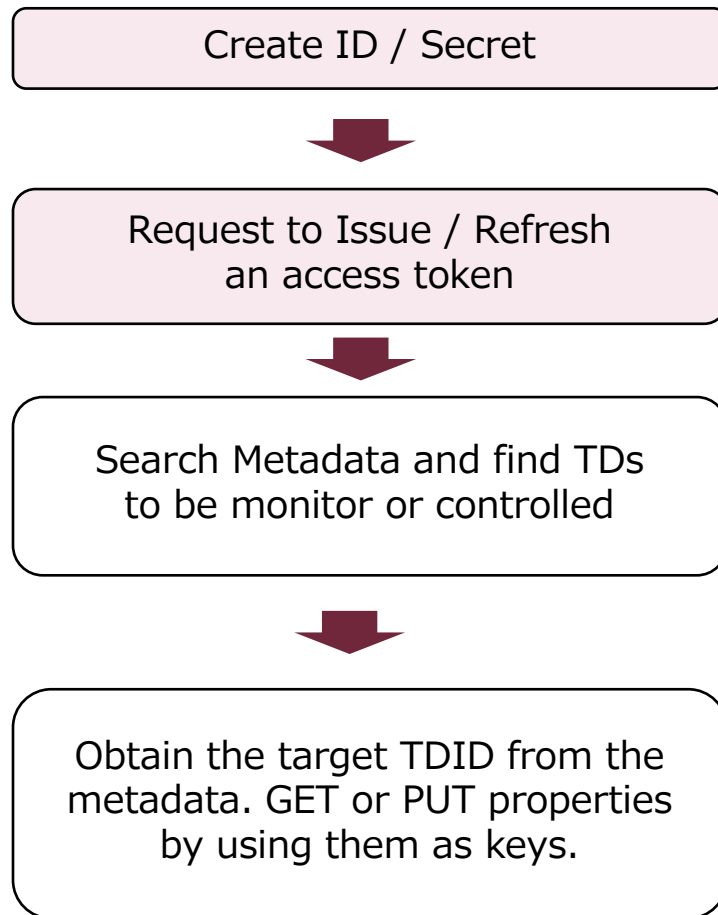
```javascript
/*Futaba librarryを利用したスクリプト*/

const futaba = require('./futaba.js');
const fs = require('fs');


let client = new futaba();
// コンフィグ（ID、シークレット、アクセストークンほか）
let obj = JSON.parse(fs.readFileSync('./data/config.json', 'utf8'));


// トークンの発行・更新
client.getAccessToken(obj)
  .then(res => {
    obj.access_token = res.access_token;
    obj.refresh_token = res.refresh_token;
    if (obj.access_token && obj.refresh_token) {
      fs.writeFileSync('./data/config.json', JSON.stringify(obj));
    }
    console.log(res);
  });
```

WoT API (2) Search Metadata

Create ID / Secret

↓

Request to Issue / Refresh
an access token

↓

Search Metadata and find TDs
to be monitor or controlled

↓

Obtain the target TDID from the
metadata. GET or PUT
properties by using them as keys.
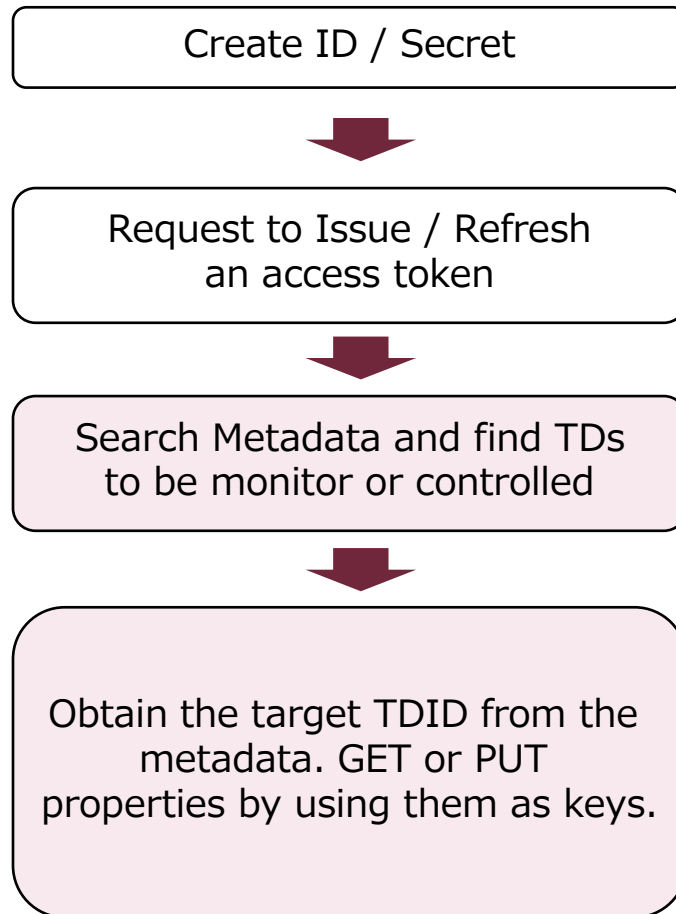
```javascript
// 特定のTD(Thing Description)を検索し、プロパティを表示
let data = {
  building: 'R90/research',
  // odata または SPARQLで検索が可能
  query_type: 'odata',
  query: "$filter=startswith(title, '環境')" //Titleに合致するthingを検索
};

// 上記の検索クエリを用いて、TDのメタデータを検索
client.getThingsWithQuery(data)
  .then(res => {
    // 取得したメタデータ (TD ID) を元にプロパティの表示
    res.things.map(item => {
      client.getThingsPropertiesWithAlias(item.tdId)
        .then(d => {
          console.log(item.title);
          console.log(d)
        });
    })
  });
```

WoT API (3) Retrieve data

Create ID / Secret

⬇

Request to Issue / Refresh an access token

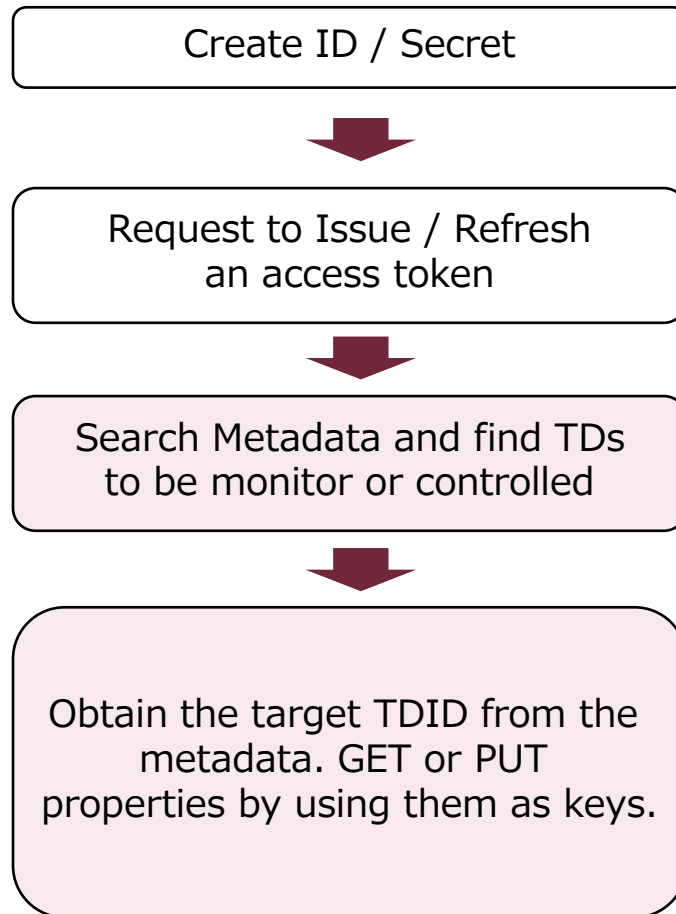⬇

Search Metadata and find TDs to be monitor or controlled

⬇

Obtain the target TDID from the metadata. GET or PUT properties by using them as keys.

title : "環境センサ:環境センサ:18254163"
titles
created : "2020-10-09T02:41:20.875462"
modified : "2020-11-09T01:53:00.439324"
securityDefinitions
security : "nosec_sc"
base : "https://futaba-dev-app-hot.azurewebsites.net/api/things/37f581e2-8b5e-4350-8a71-f82a6ecf1197"
properties
    SoundPressure
    eCO2
    eTVOC
    HighSoundPressure
    Humidity
    Light
    LowSoundPressure
    MaximumSoundPressure
    MedianSoundPressure
    MinimumSoundPressure
    Temperature
        descriptions
        forms
        writeOnly : false
        readOnly : true
        unit : "degC"
        type : "number"
        minimum : 0
        maximum : 100

環境センサ:環境センサ:18254160
{ properties:
    { SoundPressure:
        { event_time: '2020-12-09T13:54:16.708965+00:00',
          values: 38.24,
          pointId: 'R90_011402' },
    eCO2:
        { event_time: '2020-12-09T13:54:14.7192082+00:00',
          values: 546,
          pointId: 'R90_011393' },
    eTVOC:
        { event_time: '2020-12-09T13:54:12.7602548+00:00',
          values: 22,
          pointId: 'R90_011394' },
    HighSoundPressure:
        { event_time: '2020-12-09T13:54:16.7111731+00:00',
          values: 34.96,
          pointId: 'R90_011395' },
    Humidity:
        { event_time: '2020-12-09T13:54:12.929545+00:00',
          values: 51.02,
          pointId: 'R90_011396' },
    Light:
        { event_time: '2020-12-09T13:54:13.444898+00:00',
          values: 6,
          pointId: 'R90_011397' },
    LowSoundPressure:
        { event_time: '2020-12-09T13:54:16.7118353+00:00',
          values: 33.19,
          pointId: 'R90_011398' },

WoT API (4) Retrieve data

```
Create ID / Secret
```

⬇

```
Request to Issue / Refresh
an access token
```

⬇

```
Search Metadata and find TDs
to be monitor or controlled
```

⬇

```
Obtain the target TDID from the
metadata. GET or PUT
properties by using them as keys.
```

```
AHP-1-1
{ properties:
    { Status:
        { event_time: '2020-12-02T22:06:45.0397985+00:00',
          values: 0,
          pointId: 'R90_000706' },
      Failure_Alarm: { event_time: null, values: null, pointId: 'R90_000701' },
      Emergency_Alarm: { event_time: null, values: null, pointId: 'R90_000702' },
      Temperature_Out:
        { event_time: '2020-12-03T09:03:08.0553487+00:00',
          values: 33.3,
          pointId: 'R90_000703' },
      Temperature_In:
        { event_time: '2020-12-03T09:03:08.5258212+00:00',
          values: 17.8,
          pointId: 'R90_000704' },
      Electricity_Cum:
        { event_time: '2020-12-03T09:02:48.4057225+00:00',
          values: 277958,
          pointId: 'R90_000705' },
      Integrated_Flow_Rate: { event_time: null, values: null, pointId: 'R90_000707' },
      Water_Flow_Inst:
        { event_time: '2020-12-03T09:03:09.8635685+00:00',
          values: 0,
          pointId: 'R90_000708' },
      Calorie_Calc:
        { event_time: '2020-12-03T09:03:08.0350386+00:00',
          values: 0,
          pointId: 'R90_000709' } },
  correlation_id: '800005fa-0400-9600-b63f-84710c7967bb' }
```

Search for air conditioner (AHP) and get the result of the property

END

kasuya.takashi@takenaka.co.jp

kasuya@hongo.wide.ad.jp