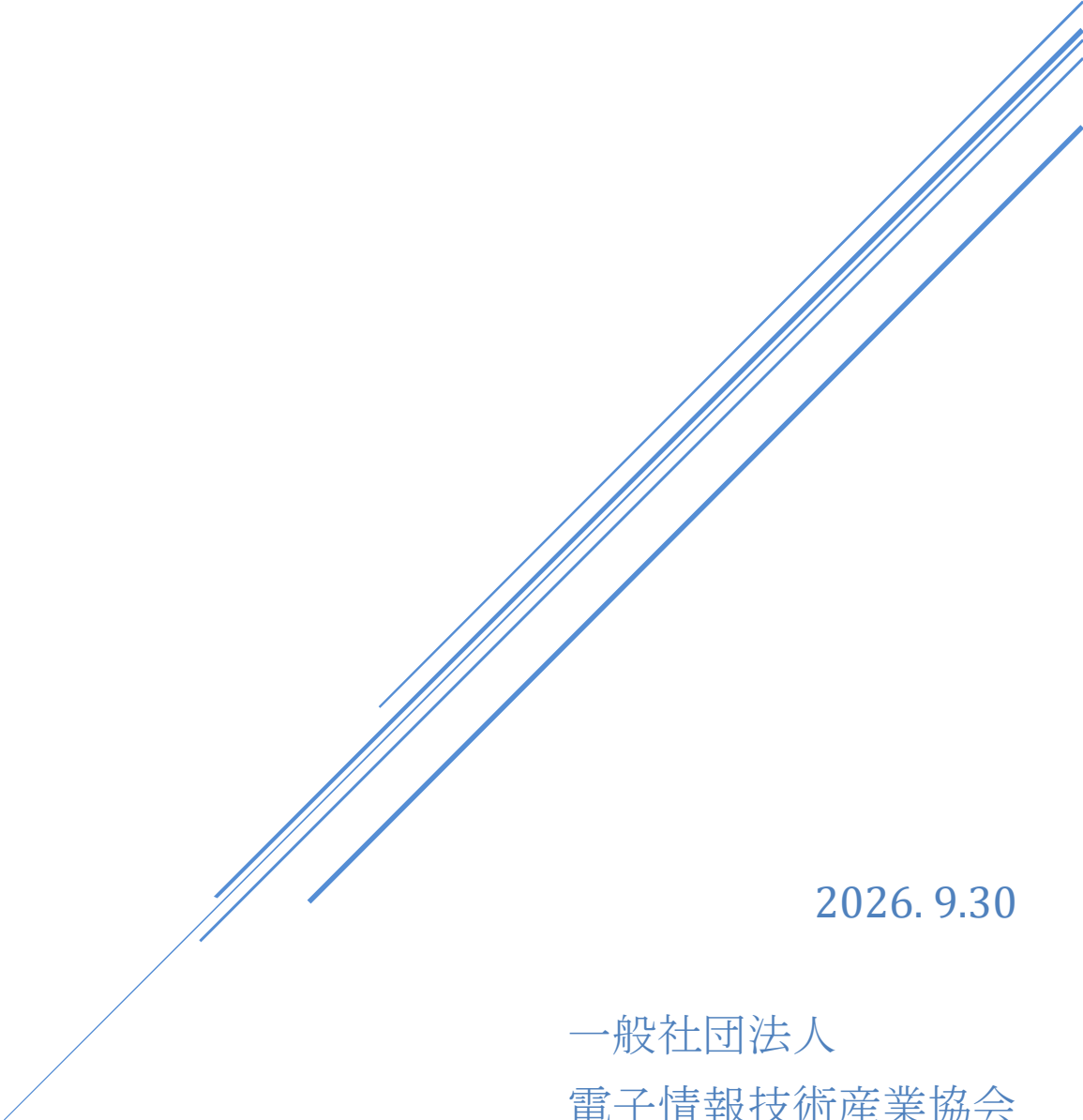


イエナカデータ連携基盤

仕様書

1.0.3 版



2026. 9.30

一般社団法人
電子情報技術産業協会
スマートホーム部会

変更履歴

1.0.3 版	公開初版

目次

1. はじめに	4
1.1. イエナカデータ連携基盤の背景	4
2. システム概要	5
2.1. イエナカデータ連携基盤の特長	5
2.2. イエナカデータ連携基盤の効果	5
3. 共通定義	7
3.1. データフォーマット	7
4. ID の種類と ID 連携	7
4.1. ID の種類	7
4.2. 上位サービスとイエナカデータ連携基盤の ID 連携	7
4.3. イエナカデータ連携基盤とメーカー間の ID 連携	8
5. データ構造	8
5.1. ユーザーのパーソナルデータ	9
5.2. イエナカデータ連携基盤ユーザー情報	9
5.3. イエ情報	10
5.4. デバイス情報	11
5.5. 機器のセンサデータ	11
5.6. 高次化データ	12
6. アーキテクチャと各ブロックの内容	12
6.1.3 層構造とブロック間の構造	12
6.2. ブロック A : Service Interface Layer	14
6.3. ブロック B : 高次化処理モジュール	16
6.4. ブロック C : 正規化・データベース	17
6.5. ブロック D : デバイス管理	18
6.6. ブロック E : ユーザー／イエ ID 管理	19

6.7. ブロック F : ECHONET Lite Web API 方式のデバイス側接続層	20
6.8. ブロック G : 専用のデバイス側接続層	22
6.9. ブロック H : カタログ	23
7. 非機能要件	23
7.1. 稼働品質・保守運用条件に関する基本方針	23
7.2. セキュリティに関する基本方針	23
7.3. プライバシーポリシーおよび利用規約準拠	24
8. 参考文献	25

1. はじめに

1.1. イエナカデータ連携基盤の背景

「スマートホーム」とは、IoT 家電・住宅設備等により収集される消費者の生活データを活用し、利便性向上や社会課題解決に繋がる様々なサービスを生み出す仕組みであり、JEITA では、2017 年より市場創出に向けスマートホーム部会を設置し、活動を進めてきた。

当初「スマートホーム」は、住宅内に設置された家電製品や設備機器を簡単に操作・遠隔から制御できるようにするホームオートメーションの領域が中心だったが、最近では、外部のサービス事業者と連携し、住宅全体の価値を高める居住者の生活を支援するサービスへと進化しつつある。さらには、消費者へ快適な生活を提供するだけに留まらず、人手不足が深刻化している不動産管理など、事業者向けのスマートホームサービスの導入も進んでいる。

その背景としては、少子高齢化に起因する人口減少による新築住宅の需要が落ち込み予測、自然災害が多い国であることからの防災レジリエンスの必要性、地球温暖化によるカーボンニュートラル対策などの社会課題の深刻化があげられる。このため、住宅に関わるさまざまなプレイヤーが協力しながら「スマートホーム」サービスの需要をいかに創出し、そのために必要なデータを扱うことができる連携基盤をいかに早急に整備していくかがカギを握ることになる。

これまで、IoT 機能付きの家電製品や設備機器は、各メーカーで稼働データを収集してクラウドサーバーで管理を行ってきた。それらは各メーカーが独自で開発・運用を行ってきたため、取得したデータは自社で囲い込んだ形で使用され、他のライバル企業に提供しないことが、データ連携サービスの開発が進まない要因となっていた。また、宅内に設置した家電/住宅設備などの機器情報もメーカーごとにバラバラであり、管理のための ID も共通的に使われる整備がされておらず、家単位でデータを連携してサービスを提供することも難しい状況になっている。

上記のような状況を踏まえ、JEITA では、スマートホームの標準通信規格を推進する団体である一般社団法人 エコネットコンソーシアムと連携し、「新サービス創造データ連携基盤検討会」を設置、「イエナカデータ連携基盤」について検討を開始した。当該データ連携基盤の開発を推進することにより、家を特定する「イエ ID」によって、家に設置された異なるメーカーの IoT 機器データを連携できるようにするとともに、マルチベンダーが生み出すデータを分析・高次化する機能を実現し、スマートホームに関連するステークホルダーに、消費者の生活データを活用しやすくするための基盤の提供を目指している。

2. システム概要

2.1. イエナカデータ連携基盤の特長

「イエナカデータ連携基盤」は、Web 上に存在する様々なサービスと、宅内の様々なメーカーの機器を効率的、かつ効果的に繋ぐ役目を担う。具体的な特長としては以下の通りであり、マルチベンダーかつ多様なサービス事業者との連携と協創を可能にする新たな「データ社会インフラシステムの実現」を目指している。

2.1.1. イエ単位の情報集約・情報生成

宅内の機器が IoT 化すると、多くの場合、メーカーごとの ID で情報が扱われるために、その宅内にある様々なメーカーの機器をひとまとめに扱う事が難しくなる。イエナカデータ連携基盤では、宅内の機器情報を「イエ単位」で扱うことが出来る様に ID を取りまとめる役割を担う。イエナカデータ連携基盤では、「イエ」とは、「宅内に住むユーザーが使用する機器からの情報を総括して扱う単位」として定義する。例えばユーザーA、ユーザーB それぞれがデバイスメーカーに登録し、イエナカデータ連携基盤に保存されたデータは、「同じイエ」のデータとして扱われる。「イエ」という単位を管理する ID として「イエ ID」を定義する。

2.1.2. 高次化データの生成

サービス事業者が求める機器情報は、多くの場合機器からの生データそのものではなく「住まい手の生活」を容易に理解可能とする「サービスで使いやすい形にしたもの」である。「イエナカデータ連携基盤」では、各機器から収集したデータをサービス事業者が使いやすい状態に加工し、要望される高次データを生成する役割を担う。

2.1.3. 機器とサービスに関する認証・認可の把握と表示および設定

イエナカデータの認可を取り扱う上で、イエナカデータ連携基盤は、どの機器の情報を、どのサービスに提供しているのかを把握できるポジションにある。そのため、これらを分かりやすく取りまとめて表示し、設定する役割を担う。

2.2. イエナカデータ連携基盤の効果

これまでの一般的なデータ連携基盤は、河川情報、交通情報など、非パーソナルな（個人特定ができない）データを扱い、これをサービス間で相互に扱えるよう接続するという部分が先行していた。その理由としては、パーソナルデータを扱う際には、

- ・認証処理、認可処理、許諾管理などを実装する必要があること
 - ・管理構造が難しいこと、個人からの許諾を適切に取得する必要があること
 - ・常時許諾状態を確認して、把握・管理できる丁寧な UI 構築が必須であること
- など、実現に非常に手間が掛かることがあげられる。

イエナカデータ連携基盤では、こうした問題を解決した個人の認証基盤を実装し、パーソナルデータのプライバシーガイドラインを守り、取り扱うことができる UI を具体的に実現することを

達成する。個人のプライバシーに配慮し、行政だけでなく、民間事業者や専門の協議会・団体、地域の大学など、あらゆるステークホルダーによる共創が実現される基盤となり得る構造で設計されている。

国際競争力維持の観点から、住宅内の IoT 家電等が生み出すイエナカデータは、機微な情報かつ産業競争力の源泉でもあるため、特定の海外企業に占有されることは、国民の暮らし情報が海外に渡るだけでなく、データ活用産業やデータの発生源であるイエナカ産業も失う懸念がある。従って、国内産業の発展に向けては、「中立性」と「サービスの持続性」を担保したデータ連携基盤の整備が重要となる。イエナカデータ連携基盤においては、従来の共通仕様として使われてきた ECHONET Lite の他、昨今注目される Matter 等の通信規格も活用できる仕組みであり、今後のイエナカ産業の発展に向け、海外製品とも連携できる基盤となっている。

3. 共通定義

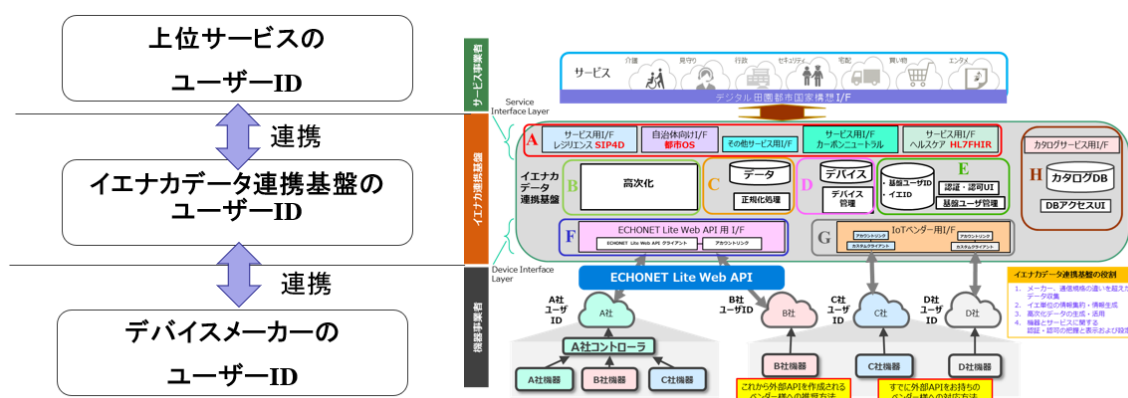
3.1. データフォーマット

- 日時形式は、RFC 3339 (ISO 8601) とする。タイムゾーンを考慮した 2026-01-02T12:34:56+09:00 の表記に従う
- リソースを表す名称は、ローワーキャメルケースとする

4. ID の種類と ID 連携

4.1. ID の種類

- 上位サービスのユーザーID
 - イエナカデータ連携基盤ユーザーID
 - デバイスメーカーのユーザーID
- a)b)間、b)c)間でそれぞれ連携を行う。



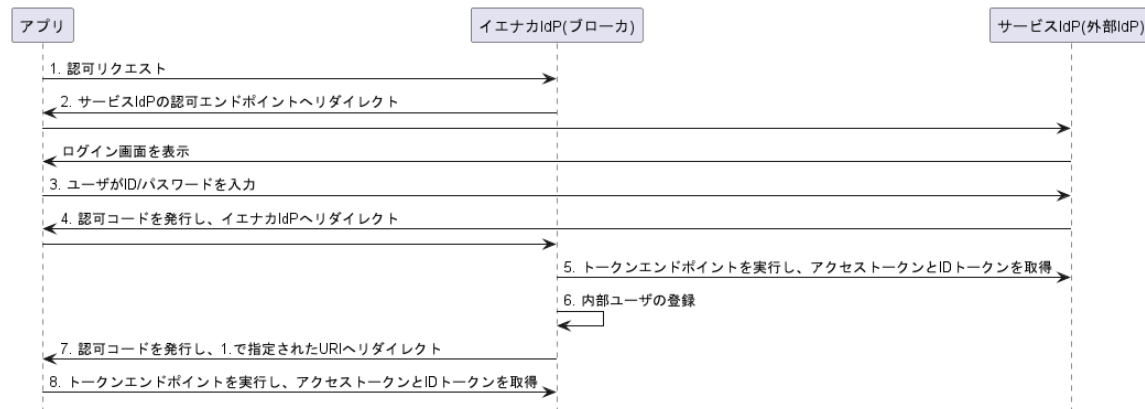
4.2. 上位サービスとイエナカデータ連携基盤の ID 連携

本節では、イエナカデータ連携基盤が自治体などの他のサービスに組み込まれて利用され、そのサービスの Identity Provider(IdP)と連携し自サービスとして一貫した認証・認可基盤を構築するための設計・実装上の要点を記載する。イエナカデータ連携基盤の IdP および自治体などの他のサービスの IdP は OpenID Connect (OIDC) をサポートすることを前提とする。

イエナカデータ連携基盤が自治体などの他のサービスに組み込まれて利用される場合、イエナカデータ連携基盤の IdP は IdP ブローキング (Identity Provider Brokering / Federation) として機能し、アプリケーションはブローカーのみを信用しサービスの IdP は外部 IdP(external IdP)として存在を意識しない構成をとる。アプリケーションが外部 IdP と直接 OIDC 連携する場合、IdP ごとの差異 (Claim 名、仕様、制限) がアプリに漏れる、IdP 追加・変更時にアプリ改修が

必要となる、認証ポリシー（MFA 等）が分散する問題が顕在化するため、IdP ブローカリングにより解決する。

IdP ブローカリングの認可コードフローは下記となる。



1. アプリはイエナカ IdP へ認可リクエストを送信
2. イエナカ IdP は外部 IdP へリダイレクト
3. ユーザーが外部 IdP の ID/パスワードを入力
4. 外部 IdP はイエナカ IdP へ認証結果として認可コードを発行しリダイレクト
5. イエナカ IdP はトークンエンドポイントを実行し、アクセストークンと ID トークンを検証
6. イエナカデータ連携基盤の内部ユーザーの登録
7. イエナカ IdP が 自ドメインの認可コードを発行しアプリヘリダイレクト
8. アプリはトークンエンドポイントを実行し、アクセストークンと ID トークンを検証

イエナカデータ連携基盤では、外部 IdP の識別子（OIDC の sub 等）を直接内部ユーザーID(イエナカデータ連携基盤ユーザーID)として利用せず属性として保持し、イエナカデータ連携基盤ユーザーID は独自に採番する。外部 IdP が発行する識別子は、その IdP のスコープ内でのみ一意であり、IdP を切り替えると ID が変わる、同一人物が複数 IdP を使うと別人扱いになる、IdP 側の仕様変更・テナント移行の影響を受ける問題を内包するため、外部 IdP が発行した識別子の利用は控える。

4.3. イエナカデータ連携基盤とメーカー間の ID 連携

イエナカデータ連携基盤からメーカーのクラウドにアカウントリンクすることで連携する。詳細は、ブロック F、及び、ブロック G の説明を参照のこと。

5. データ構造

イエナカデータ連携基盤で用いる各種データ構造を下記に記す。イエナカデータ連携基盤は、上位サービスとして複数のサービスに対応することができるが、前章記載のように、外部 IdP が異なると異なるユーザーとして判断することに留意すること。

5.1. ユーザーのパーソナルデータ

ユーザー基本 4 情報を示す。4 情報いずれも上位サービスの IdP が保持するマイナンバーカードの券面情報を利用する。4 情報の中で住所は必須とし、その他は上位サービス次第とする。現在のイエナカデータ連携基盤は、1 ユーザーが複数の住所を持つ構成は想定していない。

- 氏名
- 住所
イエの一致判別に使用され得る文字列。表記の揺れはマイナンバーカードの券面に依存する。
- 性別
- 生年月日

5.2. イエナカデータ連携基盤ユーザー情報

イエナカデータ連携基盤のユーザーに関する情報について記述する。
本情報は、下記ブロック E に保持される。

保持データ

- イエナカデータ連携基盤ユーザーID : イエナカデータ連携基盤で採番・生成した ID
- 外部 IdP に関連する情報
 - 外部 IdP 区別情報 : どの外部 IdP 由来のユーザーかを示す情報
 - 外部 IdP の識別子 (OIDC の sub 等)
 - 外部 IdP のアクセストークン、リフレッシュトークン、ID トークン
 - ユーザーのパーソナルデータ基本 4 情報 : 外部 IdP から得られる情報
- 上位サービスへの許諾情報 : scope のリスト
- イエ ID : イエの識別子
- 登録デバイスのリスト : イエナカデータ連携基盤に登録した本ユーザーのデバイスのリスト
- アカウントリンクしているメーカーに関する情報 (リスト)
 - メーカーのアクセストークン、リフレッシュトークン

scope の定義

scope はイエナカデータ連携基盤が保有するリソースデータの「何を許可するか」を定義する。イエナカデータ連携基盤が取り扱うデータや機能に応じて取り決めること。

- フォーマット： {resource}:{action}
 - resource：英小文字文字列
 - action：r もしくは w（r: 機器データや高次化データの取得、w: 機器操作の実行）
- 例
 - airconditioner:r エアコン機器データの取得
 - airconditioner:w エアコン機器の操作
 - wbgtr: 機器センサから生成された簡易 WBGT 値の取得

データの扱い

- イエナカデータ連携基盤に新規サインアップする際に生成される
- 生成のタイミングで、イエを割り当てる。割当方法はイエ情報を参照のこと
- 許諾情報は、OIDC の scope のリストの形で管理する
- イエナカデータ連携基盤でユーザーが機器を登録する度にデバイスリストが更新される

5.3. イエ情報

イエナカデータ連携基盤のイエに関する情報について記述する。
本情報は、下記ブロック E に保持される。

保持データ

- イエ ID：イエナカデータ連携基盤で生成した ID
- 住所
- 属するイエナカデータ連携基盤ユーザーID のリスト

データの扱い

- イエ ID は事前に発行する場合と、ユーザー生成時に生成する場合に分かれる
- 事前にイエ ID を発行する場合は、ユーザーがサービス利用開始前にイエ ID を生成してイエナカデータ連携基盤ユーザーID と紐づける。住所は事前発行の際にユーザーが入力する

- ユーザー生成時に生成する場合は、ユーザーの基本 4 情報の住所のような住所情報を利用して、同一住所のイエナカデータ連携基盤ユーザーID が存在すればその既存ユーザーのイエ ID に紐づける、存在しなければ新規にイエ ID を発行し紐づける
- 異なるイエが同じ住所情報を持つことはない
- 同一のイエに複数のユーザーを紐づける場合、ユーザー間で機器の情報が共有されることの許諾を求めること

ユーザー情報とイエ情報の関係

- ユーザーは必ず 1 つのイエに所属する
- 1 つのイエには、そのイエと同じ住所情報を持つ（複数の）ユーザーが所属する

5.4. デバイス情報

イエナカデータ連携基盤のデバイスに関する情報について記述する。
本情報は、下記ブロック D に保持される。

保持データ

- 機器 ID: イエナカデータ連携基盤が発行した ID （例：0x1234ABCD）
- 機器所有者のイエナカデータ連携基盤ユーザーID
- メーカーID : メーカーを区別する情報
- メーカーが発行した機器 ID
- 機器の種類
- 機器の名称（例：ユーザが設定した機器のニックネーム）
- イエナカデータ連携基盤への登録日

データの扱い

- デバイスはイエではなく、ユーザーに紐づく形で管理される。そのため、同一のイエに複数のユーザーを紐づける場合で、複数のユーザーが同一のデバイスを登録している場合、イエナカデータ連携基盤上では同一のデバイスとは見なされず、イエに複数のデバイスがあると見なされる

5.5. 機器のセンサデータ

イエナカデータ連携基盤の機器の状態に関する情報について記述する。
本情報は、下記ブロック C に保持される。

保持データ

- 機器 ID
- センサデータ種別
- センサデータの値
- 正規化されたセンサデータの値
- センサデータが生成された時刻

データの扱い

- 所謂センサーに限らず、機器の種類毎に応じた現在の機器の状態も同様に保持する
- センサデータの値と、正規化されたセンサデータの値は、同時に格納される

5.6. 高次化データ

イエナカデータ連携基盤の高次化データに関する情報について記述する。
本情報は、下記ブロック C に保持される。

保持データ

- 高次化データの ID
- 高次化データの種別
- 高次化データの値
- 高次化データが生成された時刻

データの扱い

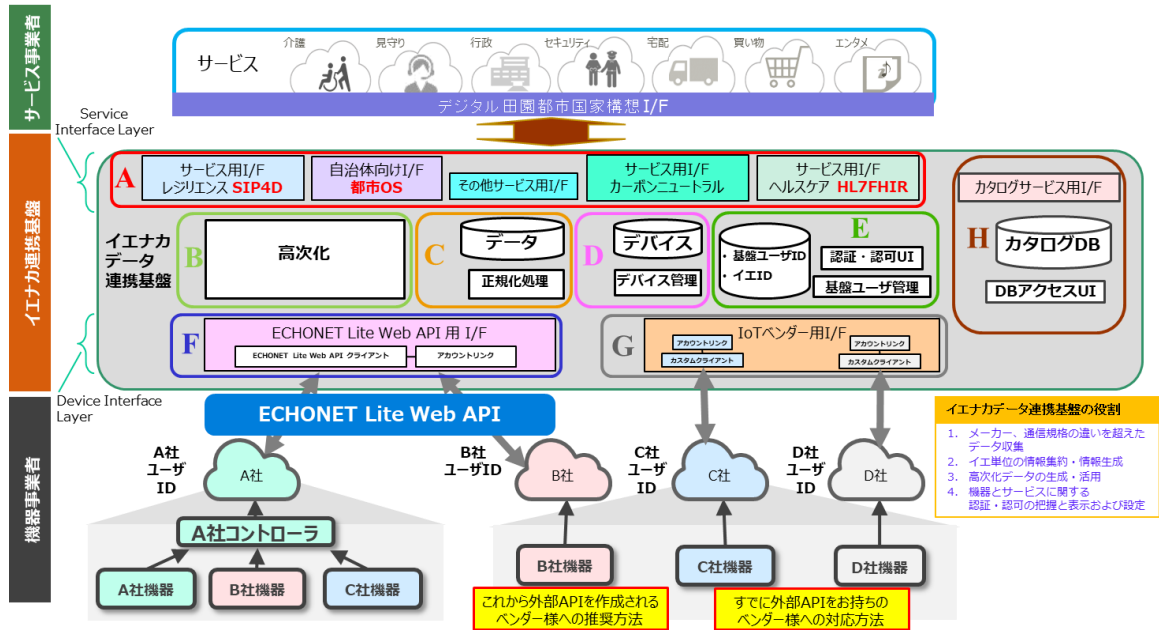
- 高次化処理モジュールにより登録される
- 高次化処理は基になる機器のセンサデータを用いて実施される。高次化処理とセンサデータの間にタイムラグが発生しうることを留意すること。高次化処理の後にセンサデータが更新されている場合が有り得る。

6. アーキテクチャと各ブロックの内容

6.1. 3 層構造とブロック間の構造

イエナカデータ連携基盤、および、その周辺機能の全体図を下図に示す。イエナカデータ連携基盤は、A～H までの計 9 個のブロックにて構成されており、上位から順番に、サービス事業者

対向する Service Interface Layer、内部処理層および、機器事業者に対向する Device Interface Layer の3層構造を取る。



イエナカ構造全体図

ブロック	名称	保持データ
A	Service Interface Layer	なし
B	高次化処理モジュール	高次化の中間的なデータ
C	正規化・データベース	機器のセンサデータ、高次化データ
D	デバイス管理	デバイス情報
E	ユーザー/イエ ID 管理	ユーザー情報・イエ情報
F	Device Interface Layer (ECHONET Lite Web API 用)	なし
G	Device Interface Layer (独自方式の IoT ベンダー用)	なし
H	データカタログ	(本版では未定義)

ブロック B～E が内部処理層に相当する。
 ブロック H については本版では未定義であり言及しない。

ここで各ブロックは選択式であり、イエナカデータ連携基盤の目的によって存在しない実装が有り得る。例えばサービス事業者との接続が不要であればブロック A は存在せず、高次化処理が不要であればブロック B は存在しない。

6.2. ブロック A : Service Interface Layer

ブロック A は、上位のサービス事業者との結合部の **Service Interface Layer** である。サービス事業者それぞれ独自のインタフェース仕様があり、イエナカデータ連携基盤は、そのインタフェースに基づいた接合層を持つ。現在のイエナカデータ連携基盤は、下記 2 種類に分類される機能を提供する。

①プル型：サービス事業者からイエナカデータ連携基盤へデータを取得

②Webhook 型：イエナカデータ連携基盤からサービス事業者へデータを通知

以下に Webhook 型のリクエストボディのフォーマット例を示す。プル型は現時点で未実装のため、本版では触れない。

また Webhook の登録について、現状は事前にサービス事業者との取り決めることとする。将来は API で登録できるよう拡張を考えると良い。X-API-KEY はクライアントが Web API で事前に登録できるようにしても、事前に手動発行する形でもよい。

```
POST \<クライアントが登録した任意の URL\> HTTP/1.1
```

```
Content-type: application/json
```

```
X-API-KEY: <事前に発行した値>
```

```
{
  "ienakaData": [
    {
      "ieNo": <イエ ID>,
      "events": [
        {
          "resource": <リソース URL>,
          "value": <データ>,
          "time": <データの発生時刻 RFC 3339>
        }
      ]
    }
  ]
}
```

通知されるデータは以下の例となる。

```
POST <クライアントが登録した任意の URL>/v1/notifications
```

```
Content-type: application/json
```

X-API-KEY: abcd123456

```
{
  "ienakaData": [
    {
      "ieNo": "fbe6a943-38f7-42e3-b808-915c9d371c5a",
      "events": [
        {
          "resource": "/devices/116ac40e-a6c3-46c2-94b5-6b5efd900ec9/properties/operationStatus",
          "value": true,
          "time": "2023-01-01T06:15:00.000+09:00"
        },
        {
          "resource": "/devices/116ac40e-a6c3-46c2-94b5-6b5efd900ec9/properties/roomTemperature",
          "value": 25,
          "time": "2023-01-01T06:15:00.000+09:00"
        },
        {
          "resource": "/ienaka/fbe6a943-38f7-42e3-b808-915c9d371c5a/properties/atHome",
          "value": 90,
          "time": "2023-01-01T06:00:00.000+09:00"
        }
      ]
    },
    {
      "ieNo": "2bcc9197-e580-43f0-b5b9-49249554138c",
      "events": [
        {
          "resource": "/devices/b5840974-0c80-4b3f-a8e3-65009cfa50dc/properties/operationStatus",
          "value": true,
          "time": "2023-01-01T06:18:00.000+09:00"
        }
      ]
    }
  ]
}
```

上記の他 Service Interface Layer が備えるべき API について記述する。
現在までのユースケースでは必要ないため本版では詳細規定しない。

- イエ ID 作成
イエ ID は事前に発行する場合に使用する API。住所を指定する
- イエ ID リストの取得
サービス事業者がサービス横断でイエのリストを取得する API

- **イエ ID 情報取得**
サービス事業者がイエ ID を指定してそのイエの情報を取得する API。イエの情報にはユーザーのリストを含むが、ユーザーの情報として、そのサービス事業者の外部 IdP の識別子（OIDC の sub 等）を返却する。（異なるサービス事業者のユーザーは含めない）
- **住所変更**
ユーザーやイエの住所を変更する API

サービス事業者が用意すべき API についても記述する。現在までのユースケースでは必要ないため本版では詳細規定しない。

- **ユーザーのパーソナルデータ取得** パーソナルデータ、すなわち、基本 4 情報を取得できる API

各ブロックとの接合

- **上位サービスと ID 連携時**
 - ブロック E の持つユーザー情報、イエ情報を格納・参照する
- **上位サービスからのプル型のデータ取得の際**
 - ブロック E の持つユーザー情報、イエ情報を参照する
 - ブロック D の持つデバイス情報を参照する
 - ブロック C の持つセンサデータ、高次化データを参照する
- **上位サービスへの Webhook データ通知の際**
 - ブロック C の持つセンサデータ、高次化データの更新時、Webhook 対象であればブロック C から通知を受ける

将来の展望

上記までは、イエを対象としたサービス事業者からのデータ取得要求、及び、サービス事業者へのデータ通知について記載してきた。この他将来的に、例えば防災減災関連における利用を考えた場合、イエ ID を伴わない形のインターフェイス、イエを束ねる地域を指定するインターフェイス、ダウンストリーム側で注意喚起するためのインターフェイスなどが考えられる。この点はまだ正確には決まっておらず、エコネットコンソーシアムをはじめ、各種団体等と協議しながら決める予定である。

6.3. ブロック B：高次化処理モジュール

ブロック B は、データの高次化処理である。高次化処理では、高次化する元のデータをブロック C のデータベースから取得し、高次化の演算を行い、新たに生成したデータをブロック C のデータベースに保存する。1 回の高次化処理では、複数の 1 次データまたは高次化データを読み込み、新たな高次化データを生成し、その結果をデータベースに書き込む場合もあり得る。

高次化処理がそのデータ生成のために中間的なデータを記録する必要がある場合は、そのモジュール内部のデータベースを利用する。ブロック C のデータベースに中間的なデータを記載するとデータベースとして保持しているデータが不明確になるため、ブロック C のデータベースは他モジュールから扱うことができる内容のみを記録する。

現段階の高次化データは、サービス毎に生成対応しているが、今後イエナカデータ連携基盤が整備されると、既に存在している高次化データを活用してサービスが構築される段階になり、ブロック H のカタログデータ部分が、高次化データの内容を示す働きを行う。

現在はイエの単位での高次化を対象としているが、将来的には、イエを束ねる地域単位の高次化についても考えられ得る。

各ブロックとの接合

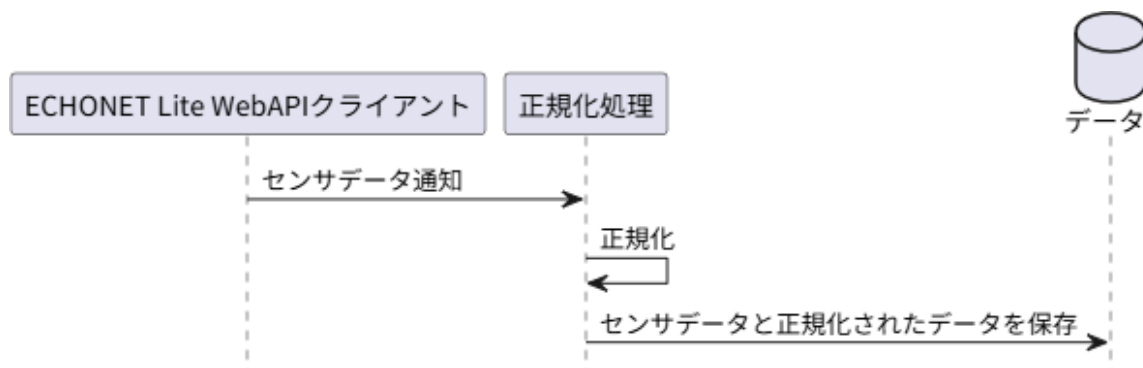
- 高次化処理時
 - ブロック C の持つセンサデータを参照する
 - ブロック C に高次化データを格納する
 - ブロック C の持つセンサデータ、高次化データの更新を受けて高次化処理がトリガーされ得る
 - イエの中にある各機器のセンサデータと連動して高次化を行うため、ブロック D のデバイス情報、ブロック E のユーザー情報、イエ情報を参照する

6.4. ブロック C : 正規化・データベース

ブロック C は、機器から取得・通知されるデータを保存する機能を持つ。正規化処理では、ブロック F や G から機器のデータを受信後、機器間でのデータの差異や、取得タイミングの違いを調整するとともに、高次化処理が扱う対象となるデータをデータベース上に保存する。

保有するデータは、5.5, 5.6 を参照のこと。

例えば、メーカーA の温度センサの単位は摂氏であるが、メーカーB の温度センサは華氏と異なる場合、正規化処理においていずれかの単位にデータを正規化する。



各ブロックとの接合

- 機器データの更新
 - ブロック F や G から機器毎のデータを受け取り格納する
- 高次化処理モジュール
 - センサデータ、高次化データの更新発生のタイミングを、ブロック B の高次化処理に知らせる
 - ブロック B よりセンサデータや高次化データの読み出し
 - ブロック B の高次化処理の結果データを受け取り格納する
- 上位サービスへの Webhook のトリガー
 - センサデータ、高次化データの更新発生のタイミングを、ブロック A の Service Interface Layer に知らせる
- 上位サービスからのプル型のデータ取得の際
 - ブロック C の持つセンサデータ、高次化データが参照される（あわせてブロック D,E が持つ情報も参照される）
- オプトアウト時
 - ブロック E からの要請に基づき、関連する機器のデータを削除する

6.5. ブロック D : デバイス管理

ブロック D は、各メーカーの機器を管理するためのデータベースと処理部である。
保有するデータは 5.4 を参照のこと。

本ブロックは、デバイスデータの管理の他、機器登録の UI や機器登録解除の機能を持つ。

各ブロックとの接合

- 機器登録時
 - ブロック F にイエナカデータ連携基盤ユーザーID、メーカーを指定して機器登録を依頼する
 - ブロック F より登録した機器の情報を受け取り、登録管理する
 - ブロック E のもつユーザーのデバイスリストに追加する
- 機器登録解除時
 - ブロック F に機器登録解除を伝え、webhook の callback 解除を依頼する
- ユーザー削除時

- ブロック E からの依頼に伴い、関連する機器をすべて登録解除する
- 上位サービスからのプル型のデータ取得の際
 - ブロック D の持つデバイス情報が参照される（あわせてブロック C,E が持つ情報も参照される）

6.6. ブロック E：ユーザー／イエ ID 管理

ブロック E は、イエナカデータ連携基盤が、1つのイエとして認識する範囲を示す「イエ ID」を取り扱う。また、イエナカデータ連携基盤として利用者からの許諾管理などを行うためにユーザーを識別するイエナカデータ連携基盤ユーザーID を取り扱う。イエナカデータ連携基盤が、自治体などの他のサービスに組み込まれて利用される場合、このイエナカデータ連携基盤ユーザーID は、上位サービスのアカウントを活用する形で実装される場合もある。詳細は、4.2 を参照のこと。

ユーザー情報やイエ情報で保持するデータやその取扱いについては、5.2 及び 5.3 を参照のこと。

ブロック E では、上位サービスにブロック C または D のイエナカデータ連携基盤が保有しているリソースデータを提供する許諾も管理する。イエナカデータ連携基盤の Identity Provider(IdP) は、OpenID Connect (OIDC) をサポートし、OIDC の scope を使い「何を許可するか」を管理する。一度許諾すれば再ログイン時はスキップする。Scope はリソースのカテゴリ毎に定義する。Scope については、5.2 を参照のこと。

ブロック E の認証・認可 UI は、ブロック C のデータの第三者提供を停止するオプトアウト API を具備する。オプトアウト API が実行されると、Service Interface Layer の Webhook の停止、および、上位サービスに発行しているリフレッシュトークンの無効化を行う。また、オプトアウト API では、ブロック C の DB が保有しているデータを削除するかを指定ができ、指定されると即時データの削除を行う。データ削除時、1つのイエに複数名のユーザーが登録されている場合、高次化データは削除せず、デバイスメーカーからの機器データのみを削除対象とする。

デバイスメーカーからイエナカデータ連携基盤へのデータ提供を停止するオプトアウトは、デバイスメーカー側のアプリで提供されることを想定する。デバイスメーカー側でオプトアウトされた際、イエナカデータ連携基盤はその動作に関与しない。

各ブロックとの接合

- 上位サービスと ID 連携時
 - ブロック A と連携し、ユーザー情報、イエ情報を格納する
- ユーザー削除時
 - ブロック D に関連する機器の削除を依頼する
 - ブロック F に webhook の callback 解除を依頼する

- オプトアウト時
 - ブロック C に含まれる関連する機器のデータの削除を依頼する
- メーカートークン関連
 - ブロック F のアカウントリンク完了時、アクセストークン、リフレッシュトークンを受け取り格納する
 - アクセストークンやリフレッシュトークン更新時、アクセストークン、リフレッシュトークンを受け取り格納する
 - ブロック F よりアクセストークンやリフレッシュトークンの読み出し
 - ブロック G との関係についても、ブロック F 同様のインターフェイスを提供する
- 機器登録・解除関連
 - ブロック D より、機器登録完了時、ユーザーID に紐づくデバイスリストに機器追加する
 - ブロック D より、機器登録解除時、ユーザーID に紐づくデバイスリストから機器削除する
- 上位サービスからのプル型のデータ取得の際
 - ブロック E の持つユーザー情報、イエ情報が参照される（あわせてブロック C,D が持つ情報も参照される）
- 高次化処理時
 - ブロック B に対し、機器 ID に対応するユーザーやイエの情報を返答する

6.7. ブロック F : ECHONET Lite Web API 方式のデバイス側接続層

ブロック F は、下位の機器事業者との結合部の **Device Interface Layer** である。ECHONET Lite Web API を実装しているメーカークラウドとの結合部であり、ECHONET Lite Web API のクライアントとアカウントリンクで構成される。

アカウントリンク部についてメーカークラウドは下記の要件を満たすものとする。

- OAuth2.0 の認可コードグラントタイプをサポートする
- アクセストークンは JWT 形式とし、iss/sub/exp/iat/azp/aud/scope を含む
- 認可コード横取りの対策として **Proof Key for Code Exchange(PKCE)**を使った認可コードグラントをサポートする（推奨）

また **scope** として機器のカテゴリ毎（例：エアコン、空気清浄機など）に定義する。

- エアコン

- 機器一覧の取得：listAirConditioner
- プロパティ値の取得：getAirConditioner
- プロパティ値の設定：setAirConditioner
- 空気清浄機
 - 機器一覧の取得：listAirCleaner
 - プロパティ値の取得：getAirCleaner
 - プロパティ値の設定：setAirCleaner

認可コードグラントにより取得したアクセストークン、リフレッシュトークンはブロック E に渡され、格納管理される。また、アクセストークンの有効期限が切れた場合、アカウントリンク部がブロック E から対象ユーザーのリフレッシュトークンを取得してトークンエンドポイントを実行し、リフレッシュしたアクセストークン、リフレッシュトークンをブロック E に渡す。また、リフレッシュトークンの有効期限が切れた場合、アカウントリンク部はユーザーへ再認証を促す。

クライアント部は、アカウントリンク完了直後、アカウントリンク部を介してブロック E に保存されているアクセストークンを使い、メーカークラウドから機器情報を取得してブロック D に渡す。また、定期的にセンサデータを取得しブロック C に渡す機能を持つ。更に、メーカークラウドからの Webhook をサポートする。Webhook はアカウントリンク直後に callbackUrl を登録する。

各ブロックとの接合

- 機器登録・アカウントリンク時
 - ブロック D より機器登録依頼を受けて、メーカーに対し OAuth2.0 のアカウントリンクを実施する。このとき既に登録対象の機器種別の scope に対して認可済みの場合、アカウントリンクはスキップして、メーカークラウドから機器情報を取得する処理を行う
 - アカウントリンク完了時、ブロック E に、アクセストークン、リフレッシュトークンを格納する
 - アカウントリンク完了直後、ブロック E に保存されているアクセストークンを使い、メーカークラウドから機器情報を取得してブロック D に格納する
 - メーカークラウドに対し、登録機器の Webhook callbackUrl を登録する
- 機器データのやりとり
 - メーカークラウドからの Webhook callbackUrl を受けて、得られた機器データをブロック C に渡す

- Webhook に対応していないセンサデータについて、必要に応じて定期的にセンサデータを取得しブロック C に渡す。このときブロック E に格納されているアクセストークンを利用する

ECHONET Lite Web API インターフェイス仕様

- アカウントリンク
参考文献 1)2) 参照
- ECHONET Lite Web API
参考文献 6) 参照

API 仕様部 Version 1.2.0

- 機器一覧取得(5.6 節)
- 機器情報(Device Description)取得(5.7 節)
- 機器オブジェクトのプロパティ値操作(6.5 節)
- GET 対象プロパティ値の一括取得
- 指定プロパティの値設定
- 機器オブジェクトのプロパティ値通知 (INF) (5.10 節 Webhook による INF 実現例)
- 対象プロパティについて
イエナカデータ連携基盤の共通仕様として必須プロパティはない。イエナカデータ連携基盤の目的に応じて取り決める。1 つの家に複数存在するような機器種別の場合は、機器登録時にエンドユーザーが機器区別できるようにニックネームに対応することを推奨する。

機器の操作

イエナカデータ連携基盤が、機器の操作に対応する場合、ECHONET Lite Web API の PUT や PATCH の呼び出しを行って機器の操作を行うことも可能である。この場合、機器操作の応答に含まれる機器の状態をブロック C に引き渡す。

6.8. ブロック G：専用のデバイス側接続層

ブロック G は、ブロック F 同様に、下位の機器事業者との結合部の Device Interface Layer である。ELWA 以外の IoT 専用のデバイス側接続層であり、各メーカーの独自仕様に準ずる。アカウントリンクでは、メーカー独自のアカウントリンクの仕組みに従い、イエナカデータ連携基盤ユーザーID とメーカーのユーザーを紐づけ、アクセストークンのようなリソースアクセスに必要な情報がメーカーから提供される場合はブロック E に渡す。カスタムクライアントでは、メーカー独自のリソースアクセスの仕組みに従い、メーカークラウドから機器情報を取得できる場合は

取得してブロック D に渡す。メーカークラウドから取得できない場合はオフラインでブロック D に登録する。また、定期的にセンサデータを取得しブロック C に渡したり、Webhook 等によるメーカークラウドからのセンサデータ通知を受信しブロック C に渡したりする。

専用の独自方式を使うこと以外はブロック F と同様。他ブロックとのインターフェイスは極力ブロック F に合わせる。

6.9. ブロック H：カタログ

イエナカデータ連携基盤が保持しているデータのカタログ処理部であり、今回の対象から除外する。

7. 非機能要件

本章では、イエナカデータ連携基盤を名乗る実装が満たすべき「非機能要件に関する共通的な考え方」を示す。本章は、特定実装の稼働品質・性能・セキュリティを直接規定するものではなく、イエナカデータ連携基盤としての原則を示すものである。

7.1. 稼働品質・保守運用条件に関する基本方針

1. 本仕様書は、特定実装の稼働率・可用性・RTO/RPO・システム構成などの具体的な稼働品質指標は定義しない。
2. API のレイテンシ、単位時間内の呼び出し可能回数についても同様。
3. 本基盤を実装する事業者は、連携する上位サービスに対し、十分な稼働品質を確保する旨を自ら表明し、必要に応じて公開すること。
4. 上位サービスは複数存在し得るため、各サービスに対して異なる稼働品質水準・保守運用条件を提示してよい。
5. 本基盤の共通した指標として、利用サービスに支障をきたさない安定動作を提供する義務を負うものとする。

7.2. セキュリティに関する基本方針

1. 本仕様書は、暗号方式・脆弱性管理・ネットワーク構成などのセキュリティ要件を具体的に定義しない。
2. 本基盤を実装する事業者は、上位サービス・関連法規（例：個人情報保護法）に基づいた適切なセキュリティ対策を自ら定め、表明すること。

3. パーソナルデータを扱う場合、データ取り扱いルール、安全管理措置、第三者提供方針を含むセキュリティ設計を明確にすること。
4. **Device Interface Layer** の下層に位置するメーカークラウド領域は、各メーカーのセキュリティ方針に従うものとし、本仕様書では要件を定めない。

7.3. プライバシーポリシーおよび利用規約準拠

1. 本基盤を実装する事業者は、自社のプライバシーポリシーを明示し、利用者に対して公開すること。
2. 本基盤は複数の上位サービスおよび複数メーカーのデータを扱う特性を持つため、各サービスの利用規約・各メーカーの利用規約に準拠したデータ処理が必要となる。

8. 参考文献

- 1) OAuth2.0
<https://datatracker.ietf.org/doc/html/rfc6749>
- 2) OIDC
https://openid.net/specs/openid-connect-core-1_0.html
- 3) PKCE
<https://datatracker.ietf.org/doc/html/rfc7636>
- 4) JWT
<https://datatracker.ietf.org/doc/html/rfc7519>
- 5) RFC3339
<https://datatracker.ietf.org/doc/html/rfc3339>
- 6) ECHONET Lite Web API API 仕様部 v1.2.0
https://echonet.jp/wp/wp-content/uploads/pdf/General/Standard/web_api/ECHONET_Lite_Web_API_Specs_v1.2.0.pdf
- 7) ECHONET Lite Web API 機器仕様部 v1.6.1
https://echonet.jp/wp/wp-content/uploads/pdf/General/Standard/web_api/ECHONET_Lite_Web_API_Dev_Specs_v1.6.1.pdf