OPEN CONNECTIVITY FOUNDATION™

OCF 仕様書：ご紹介と概要

TECHNOLOGY STEERING COMMITTEE

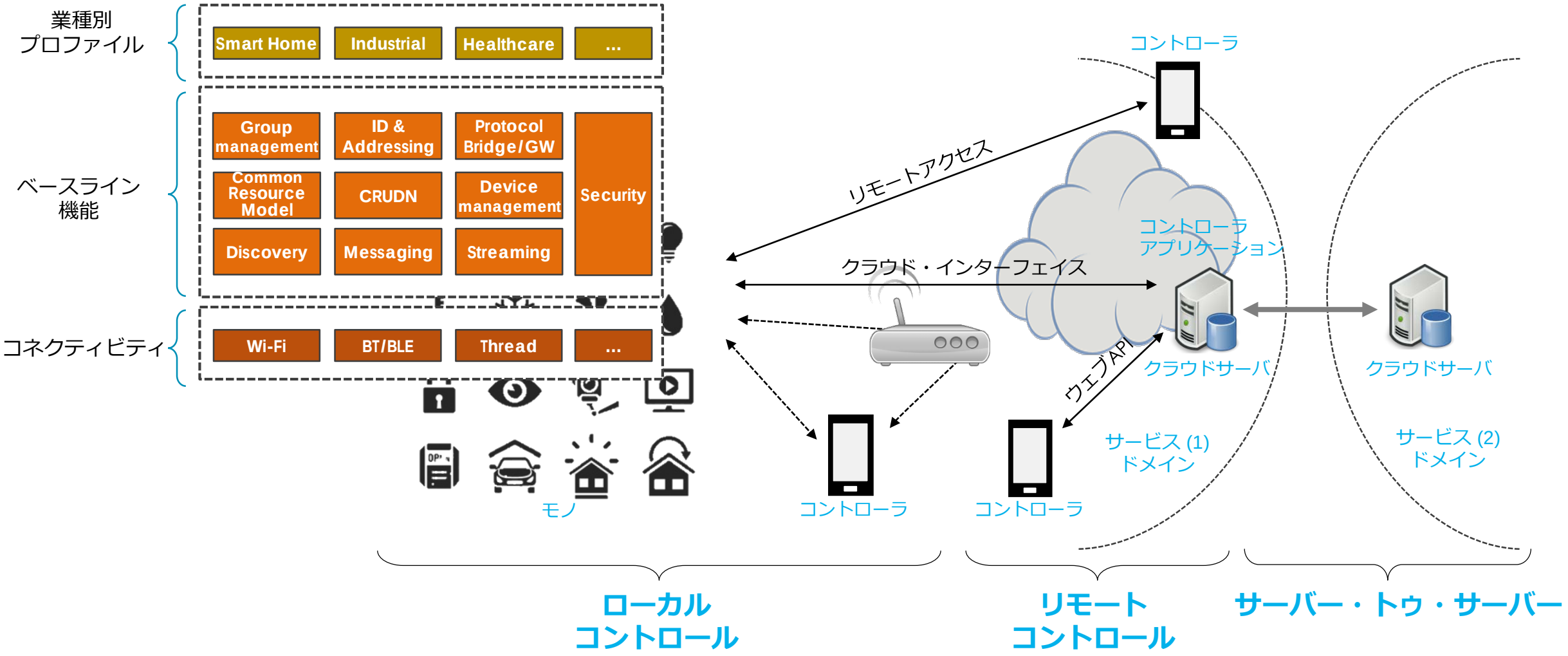
2017年07月

目次

- IoTエコシステムのテクノロジー原則
- Open Connectivity Foundationのご紹介
- OCF仕様の概要
  - コア・フレームワーク
  - セキュリティ
  - ブリッジング
  - 各種リソース
  - OCF - AllJoyn マッピング
  - スマートホームデバイス のプロファイル

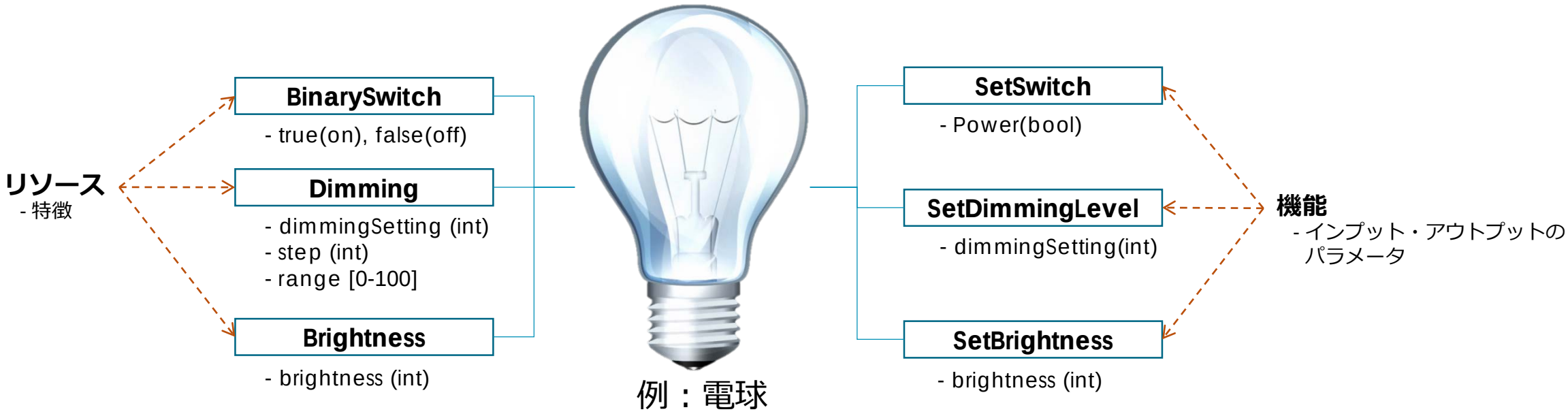
IoTエコシステムのテクノロジー原則

IoTの範囲（スコープ）



様々なモノの定義におけるアプローチ

- モノのリソースや特徴の定義
- モノの機能や役割の定義



- （no Verbs（動詞なし））＋　オブジェクト
  *トランスポートレイヤーの特定の動詞（CRUDN）を使用
- RESTful アーキテクチャにおけるリソースモデル
  （例：W3C, CSEP等）

- （Verbs（動詞）＋　オブジェクト)
- RPC モデル

Constrained Things*のサポート

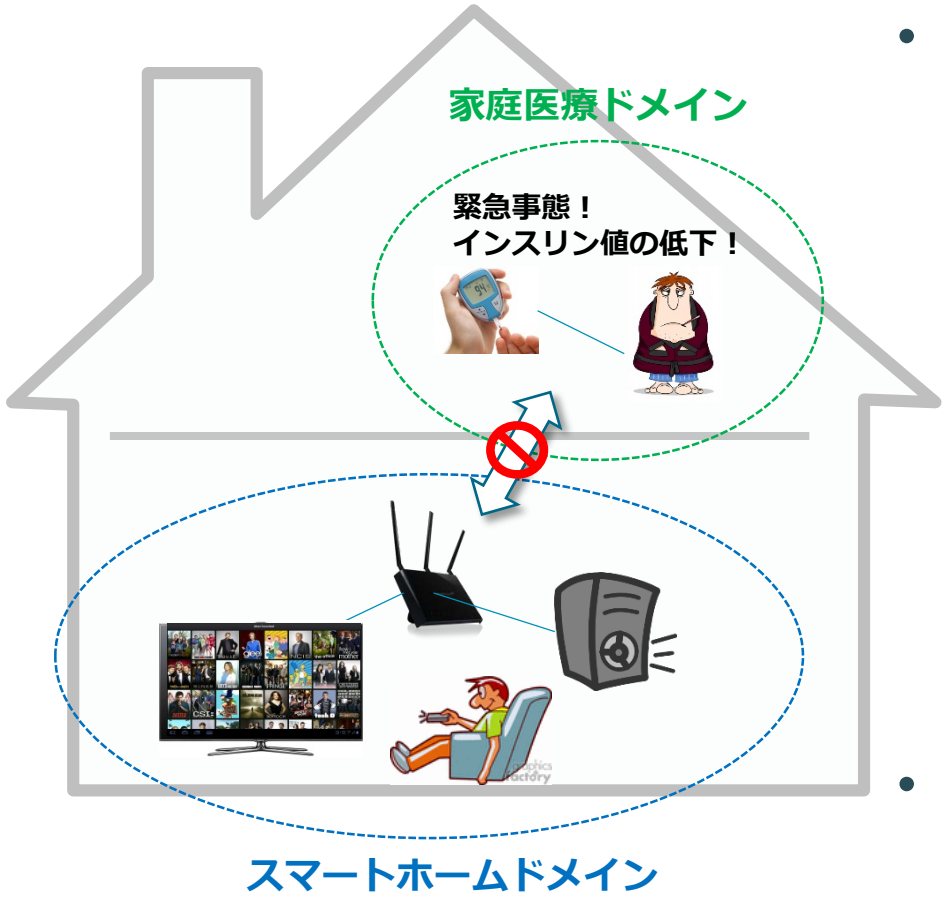
RFC 7228の定義におけるクラス2デバイス

- オーバーヘッド（負荷）およびトラフィックの低減
  - CPU ロード・メモリーへのインパクト・トラフィック・帯域幅の最小化
    - コンパクトなヘッダー
    - バイナリー・プロトコル
    - ペイロードのエンコーディングを圧縮化

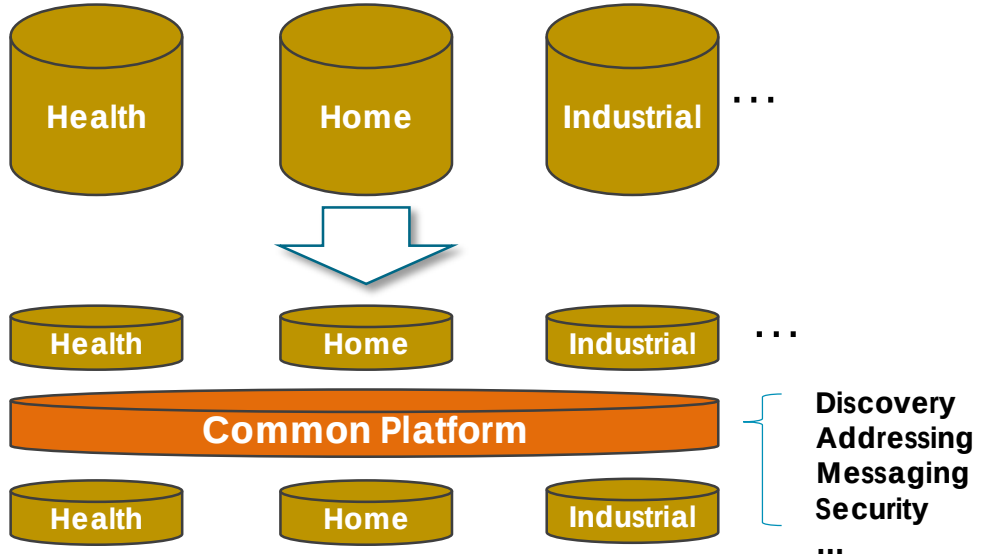
- 複雑さの低減
  - シンプルなリソースモデル
    >短い URI （リソースタイプを定義した遅延バインディング）
    >浅く広い階層

*処理能力・ストレージ・転送量等が制限されたモノ

複数の業種バーティカルをサポート



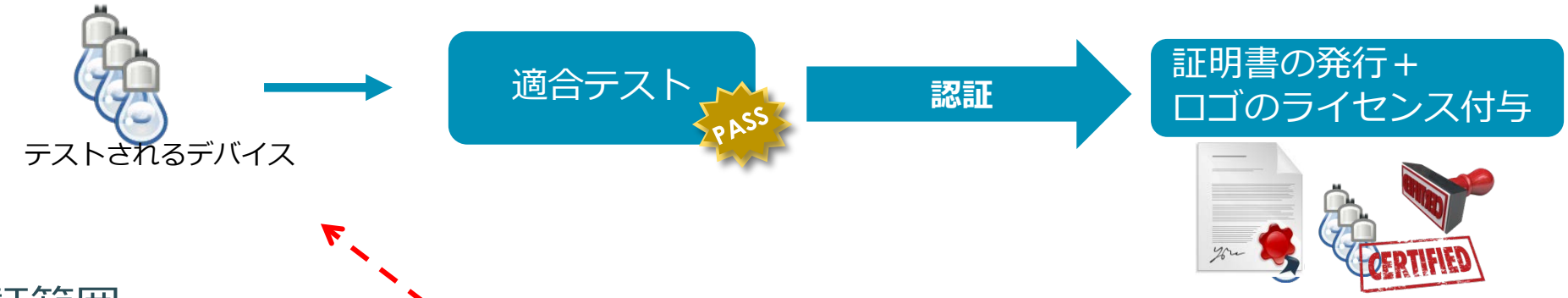
- 過去の業種バーティカルは、一般的に縦割り設計
  →統一化された相互コミュニケーション方式が存在しない



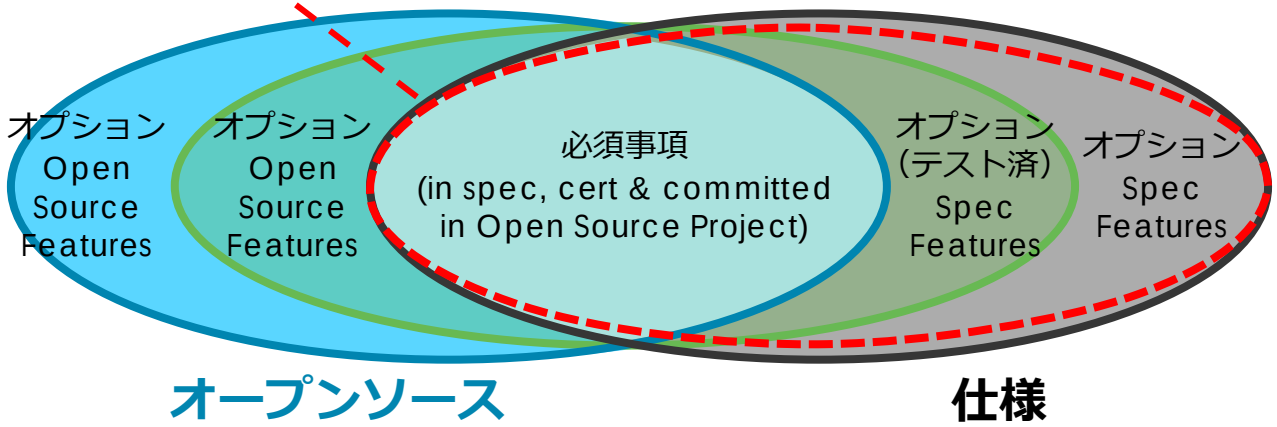
- 共通プラットフォームが共通のサービスやデータモデルを提供することにより、業種バーティカル毎に分けられた各サービス間の協業や相互作業の基盤を提供

適合・認証プロセス

- 適合テスト – それぞれのデバイスが仕様への適合性を証明



- 認証範囲



ライセンス付与

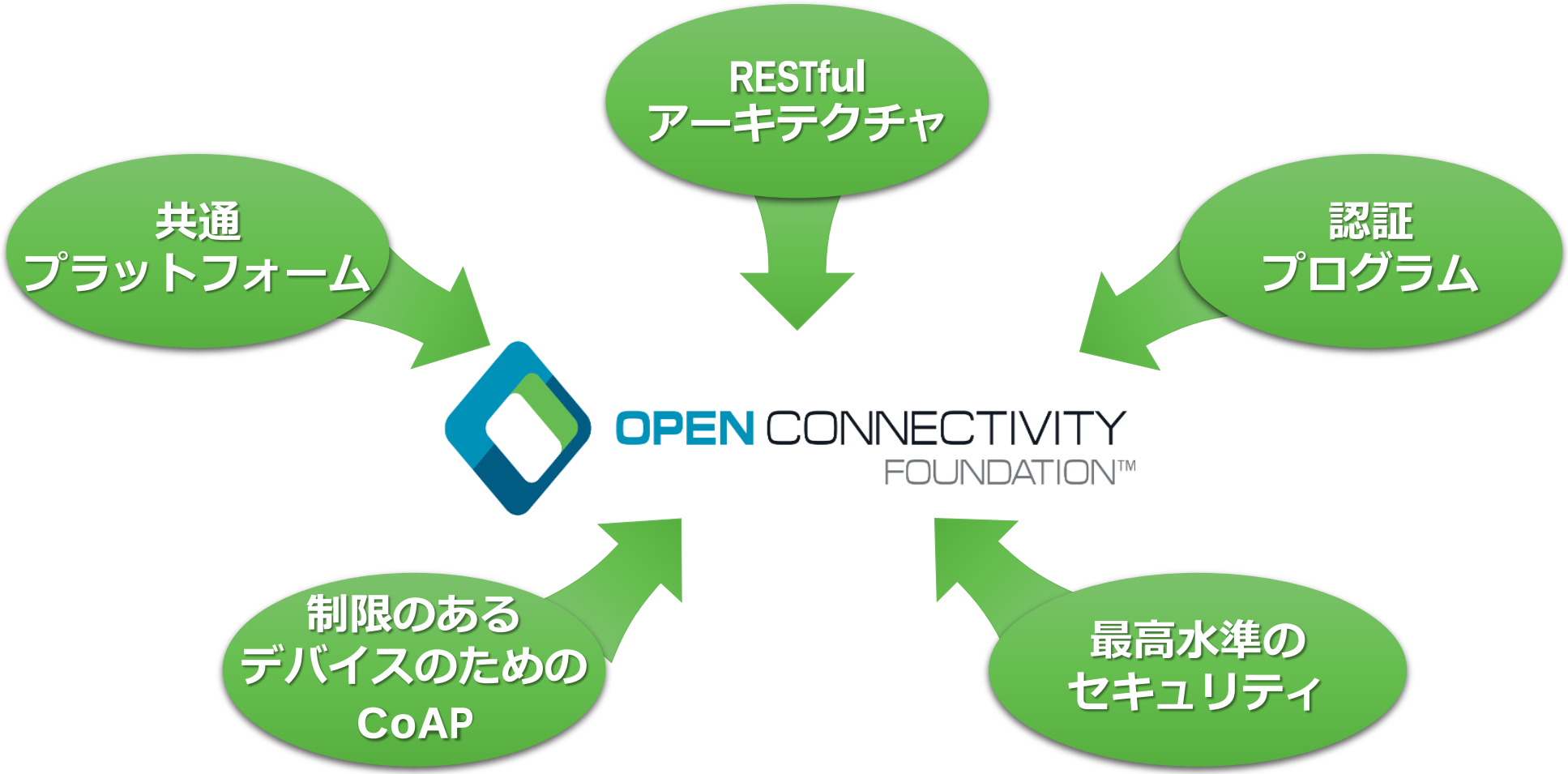
- 知的財産権（IPR）ポリシー：RAND-Z > RAND >> no IPR policy
- オープンソース： Apache 2.0 > Internet Systems Consortium (ISC)

- すべてのモノがインターネット上で繋がるというIoTの特性上、ライセンスリスクの回避はメーカーにとっての最重要課題となる
  - すべてのつながったモノが潜在的リスクにさらされる

- メーカーフレンドリーなライセンスおよび知的財産権のポリシーを提供することはスタートアップと大企業の双方にとって有益なため、市場拡大に寄与する。そのような条件を満たす知的財産権ポリシーは、明確で理解しやすいものである必要があり、またすべてのIP所有者から同様の条件が提示されていることを確認できるものでなければならない

OPEN CONNECTIVITY FOUNDATIONのご紹介

OCFのご紹介：IoTへの最適化



OCF：技術開発分野

- コア・アーキテクチャ
  - リソースの基礎フレームワーク
  - ディスカバリ
  - CRUDN
- セキュリティ
- （特定の業種バーティカルに依存しない）リソースモデル
- デバイスのプロファイル
  - スマートホーム
  - ヘルスケア
  - 自動車
- トランスポート・バインド

OCF：主要コンセプト（1/2）

- **IoTに特化・最適化したプロトコル（CoAP等）**
  - 制約のあるデバイスのために、個別の配慮・検討を行う
  - RESTfulアーキテクチャへの完全な準拠
  - ディスカバリおよびサブスクリプションのメカニズムの組み込み（ビルトイン）
- **柔軟なソリューション構築を可能にする基準やオープンソース**
  - 可能な限り広範囲なオプションにより、あらゆるデバイス、フォームファクタ、企業、および市場に対応
  - オープンソースは、課題解決におけるひとつの手段に過ぎない

OCF ：主要コンセプト（2/2）

- **相互運用性のための認証テスト**
  - デバイスが仕様に準拠している旨を確認するための正式な適合テスト
  - プロダクト間の相互運用性を確認するためのプラグフェストを開催
- **認証・ロゴ発行プログラム**
  - OCF認証ロゴマークを取得した製品は、OCFの仕様に準拠していることが保証される
  - このロゴマークが付与されている製品が、相互運用性が確保された製品群（エコシステム）の一部であることを示す

OCF仕様の概要

OCFが今後提供していくもの

標準を規定する仕様

- 次頁ご参照

リソースモデル（oneIoTaを通じて提供）

- 特定のドメインに依存しないリソース
- エコシステム・マッピングのための派生モデル
  - 現状：OCF-AllJoyn (CDM 16.4)

認証プロセス

- テストポリシー（認証プロセス要件定義書）
- テストプランおよびケーススタディ（認証テスト要件定義書）

仕様構造

インフラストラクチャ

- コア・フレームワーク
- セキュリティ
- ブリッジング
- デバイス仕様

リソースモデル

- リソース仕様（oneIoTaコンテンツを反映）
- OCFリソース・AllJoyn インターフェイス間のマッピングに関する仕様（oneIoTaコンテンツを反映）

仕様の公開について

仕様やリソースタイプの定義については、以下のURLをご参照下さい：

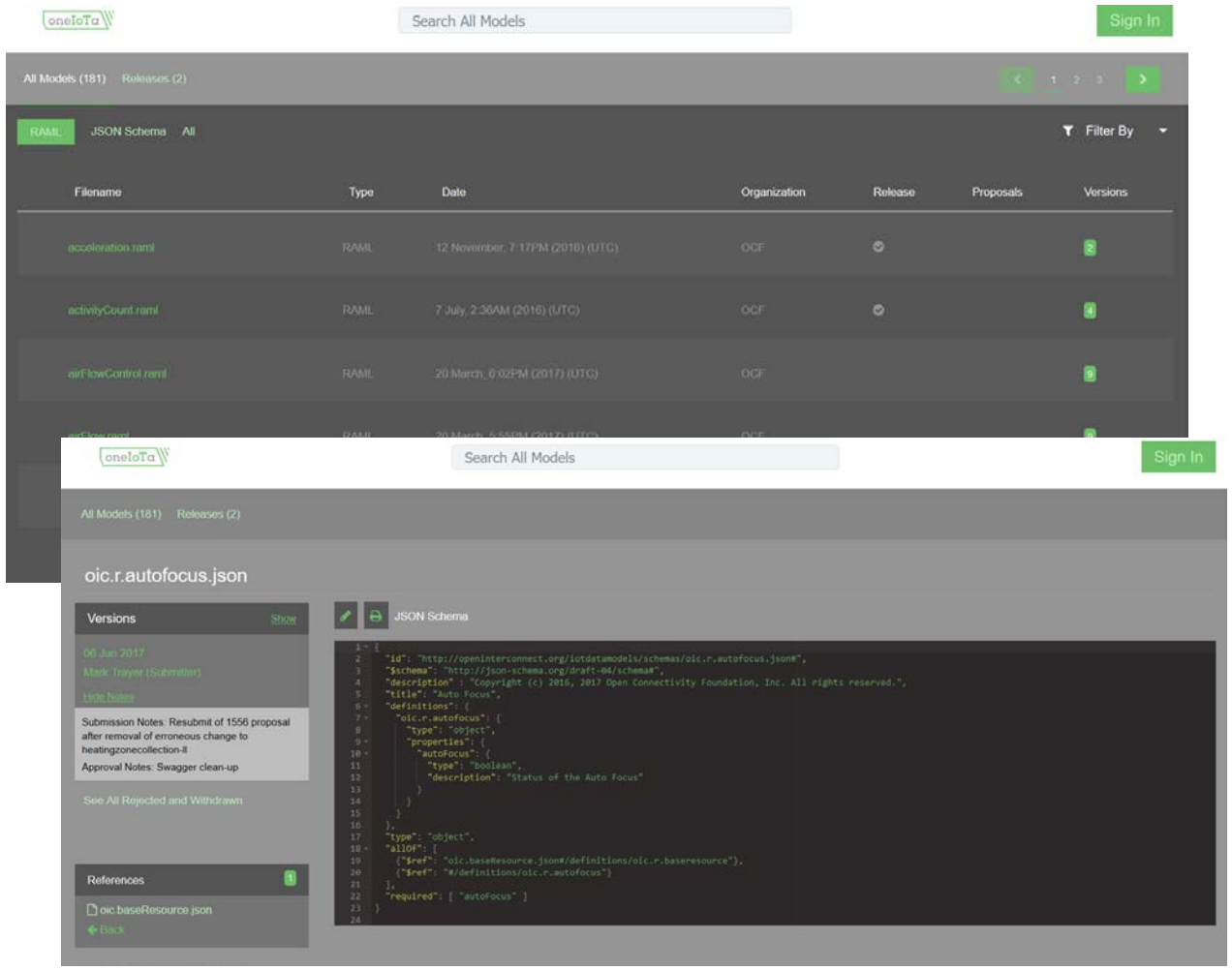
OCF 仕様：

- https://openconnectivity.org/developer/specifications

リソースタイプの定義：

- コア・リソース：https://github.com/openconnectivityfoundation/core
- ブリッジング・リソース： https://github.com/openconnectivityfoundation/bridging
- セキュリティ・リソース： https://github.com/openconnectivityfoundation/security-models
- 各業種バーティカルのリソースおよび派生モデル：https://oneiota.org/documents?filter%5Bmedia_type%5D=application%2Framl%2Byaml

oneIoTa ツール



- ウェブベースの開発ツール（http://oneiota.orgご参照）
- RAML、 JSON、および Swagger2.0 の各シンタックスをサポート
- 現状で既にすべてのOCFリソース、さらに同リソースのすべての Swagger2.0 バージョン、ならびに OCF-AllJoyn 派生モデルを搭載
- 複数の団体をサポート
  - それぞれの提供団体が、独自のライセンス条件を設定することが可能

インフラストラクチャー：コア・フレームワークの仕様

概要

コア・フレームワークに関する目標

- コア・フレームワークの仕様の範囲
  - コア・アーキテクチャのフレームワーク、メッセージング、インターフェイス、ならびに承認済みユースケースシナリオに基づくプロトコルに関する技術的仕様を定義する
  - 基礎的な相互運用性を維持したまま、コア上における業種バーティカル（例：スマートホーム）のプロファイル開発を可能にする
- リソースが制限されたデバイスからリソースが豊富なデバイスまで、スケーラブルなコア・フレームワークを設計する
- 再利用可能なオープンスタンダード・ソリューションが存在する場合は活用する（例：IETF）
- IoTivity のオープンソース・リリースとの整合性を確保する

RESTful アーキテクチャ

エンティティ
ハンドラー

物理的エンティティ

クライアント

サーバ

REST Architecture Style

- Addressable resources
- A uniform, constrained interface
- Representation based manipulation
- Communicate statelessly
- Hypermedia State Engine

リクエスト

レスポンス

CRUD & N オペレーション

```
{
  "n": "myRoomTemperature",
  "rt": "oic.r.temperature",
  "if": "oic.if.a",
  "id": "example_id_xyz",
  "temperature": 23,
  "units": "C",
  "setValue": 25
}
```

リソース（表現）

リソース（表現）

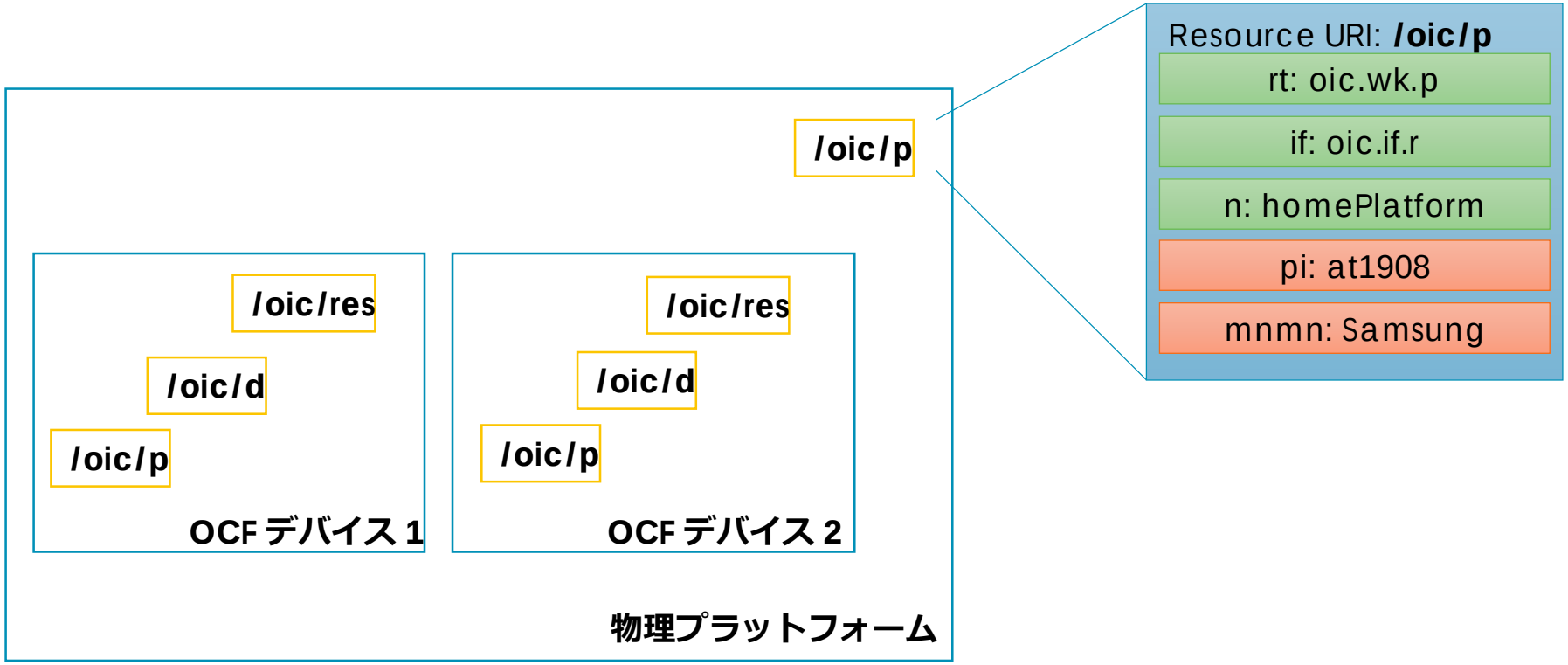
- RESTful（Representational State Transfer）アーキテクチャ
  - リソースベースのオペレーション
    - 実在する「エンティティ」は、「リソース」として表現される
  - リクエスト/レスポンスによるリソースの操作：CRUDN

OCFにおける「Role（役割）」

- 現状のOCFアーキテクチャにおいては、デバイス用に二つの論理的な役割が定義されている
  - OCFサーバ：ホストされたリソースを公開し、 ディスカバラブルであり、クライアント側から発動されたトランザクションに応答する論理エンティティ
  - OCF クライアント：ディスカバリおよびCRUDNアクションでOCFサーバ上のリソースと相互作用する論理エンティティ
- OCFデバイスは、いずれか片方もしくは双方の役割を担う

OCFデバイスの構成

- OCFデバイスのコンセプト

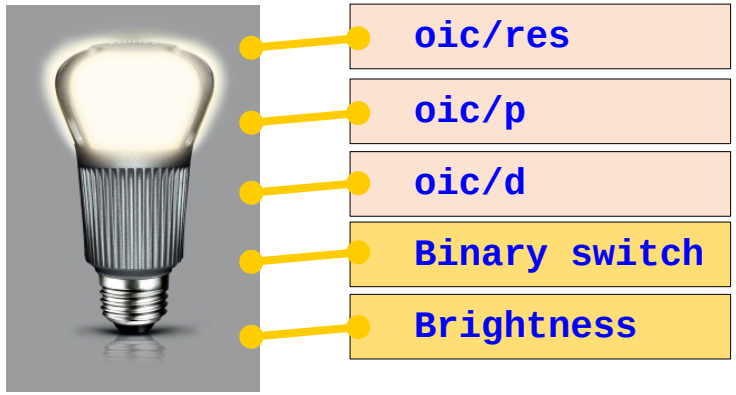


デバイス事例：照明デバイス (oic.d.light)

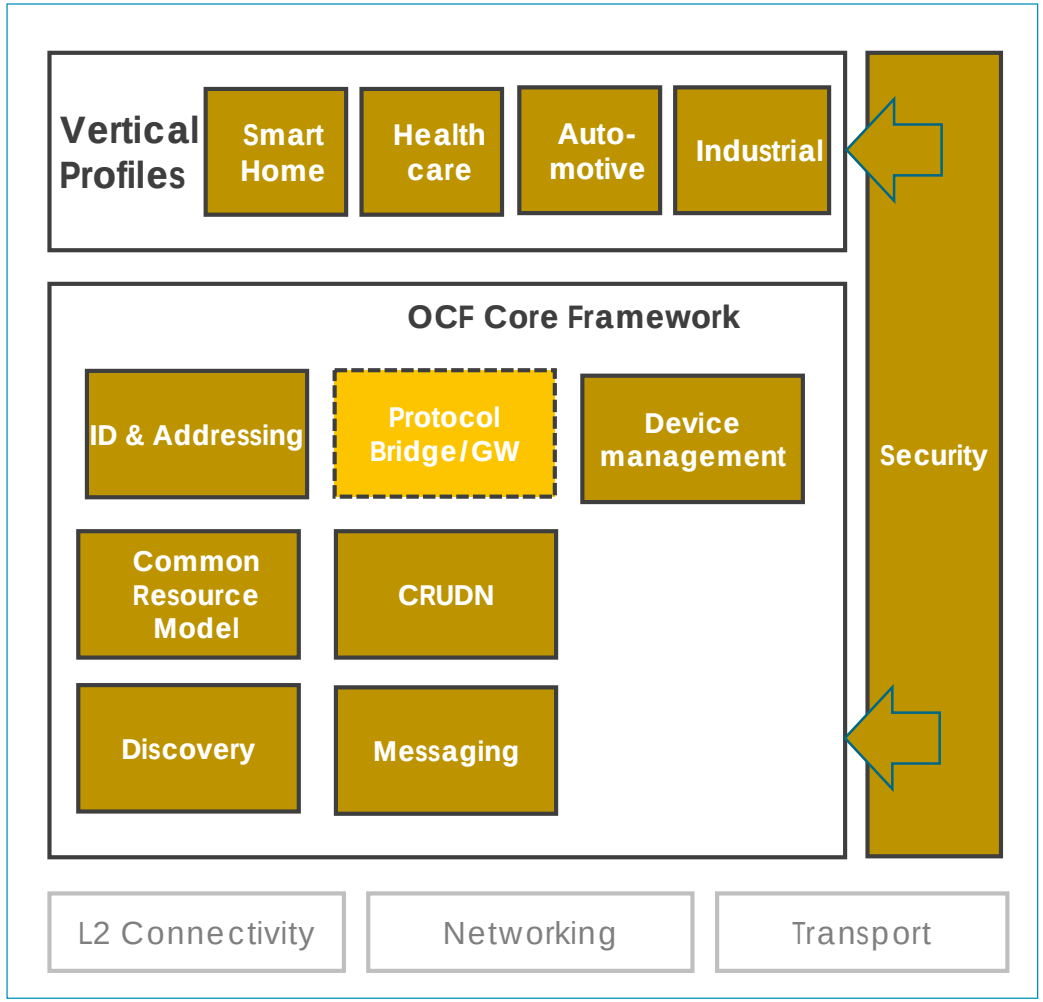
- 事例概要
  - （1）バイナリスイッチ、および（2） 輝度リソースを有するスマートな照明デバイス
- デバイスの種類：照明デバイス （oic.d.light） ［ドメインにより定義される］
- 関連リソース：
  - 必須コアリソース：oic/res, oic/p, oic/d
  - 必須セキュリティリソース：（図においては非表示）
  - デバイスに特化したリソース：バイナリスイッチ（oic.r.switch.binary）
  - その他のオプショナルなリソースも開示可能。この事例では輝度リソース（oic.r.light.brightness）

事例：スマート照明デバイス

デバイス名	デバイスの種類	関連リソースの種類	M/O
Light	oic.d.light	oic/res (oic.wk.res)	M
		oic/p (oic.wk.p)	M
		oic/d (oic.d.light)	M
		Binary switch (oic.r.switch.binary)	M
		Brightness (oic.r.light.brightness)	O



OCF仕様の特徴：コア・フレームワークの仕様



① **Discovery:** デバイスのディスカバリにおける一般的な手法（IETF CoRE）

② **Messaging:** 制限されたデバイスのサポートを標準搭載（IETF CoAP）、ならびにブリッジを通じたプロトコル変換

③ **Common Resource Model:**実在するエンティティはデータモデル（リソース）として定義

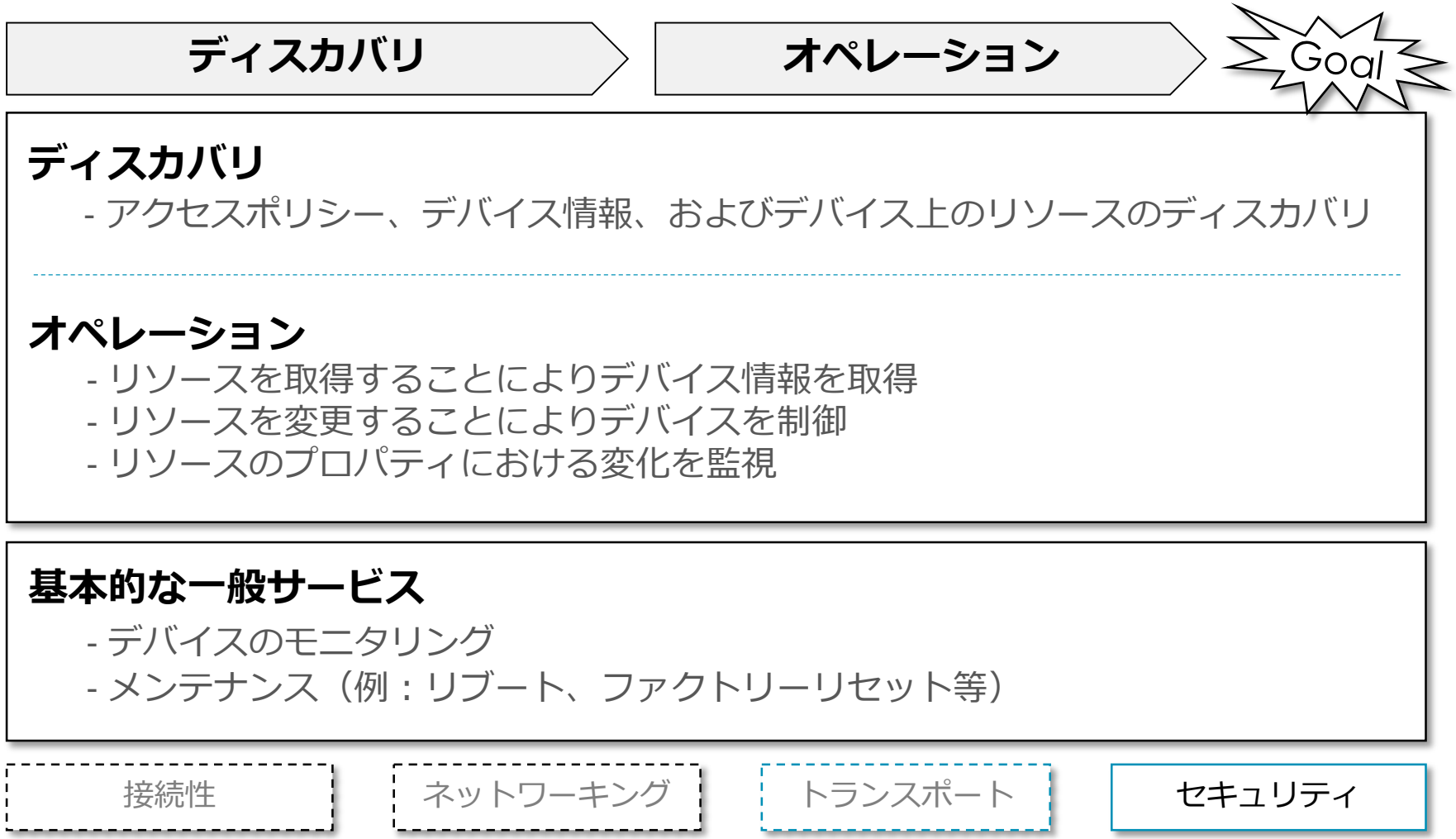
④ **CRUDN:** Create, Retrieve, Update, Delete およびNotifyのコマンドで構成される簡素なリクエスト/レスポンスのメカニズム

⑤ **ID & Addressing:** OCFのIDおよびOCF エンティティ（デバイス、クライアント、サーバ、リソース）へのアドレス割り当て

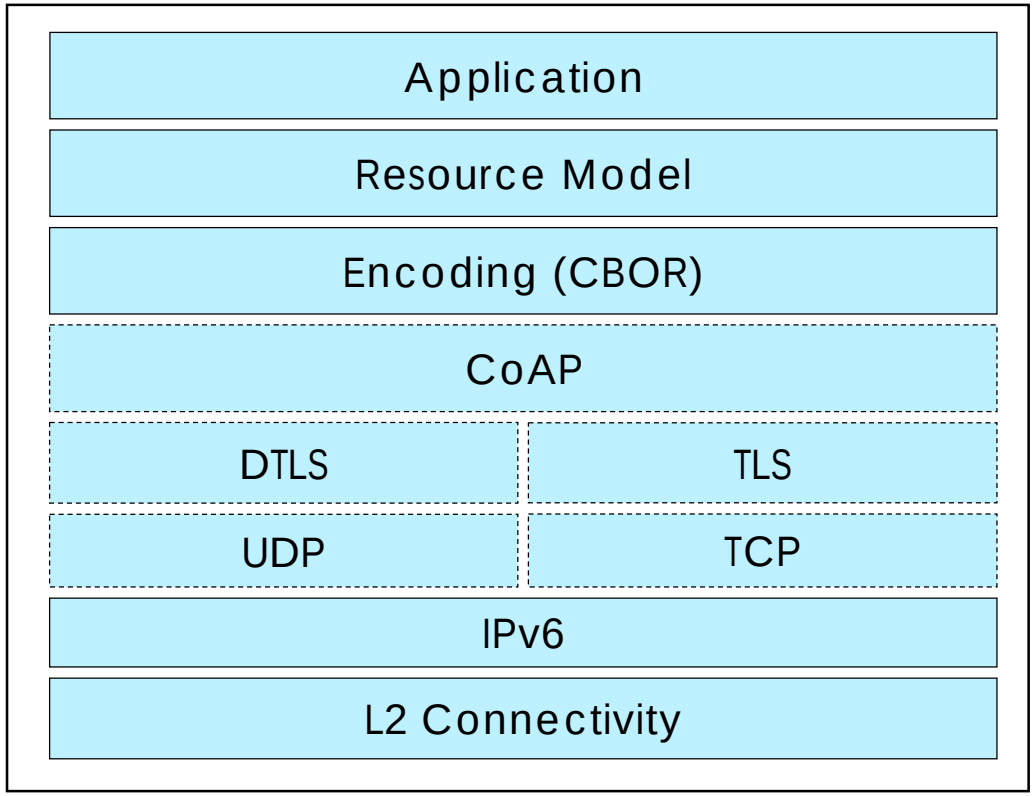
⑥ **Protocol Bridge/GW**: ブリッジングの仕様によって処理（一部、コアへの影響も存在）

OCFエコシステムにとってセキュリティは最重要項目であり、全ての要素に該当する

OCF コア・フレームワークの基本オペレーション



プロトコル・スタック



OCFスタック

エンドポイント・ディスカバリ（CoAPディスカバリ）

- OCFデバイスは、IANA によって定義された OCFサービスアドレス（デフォルトのCoAPアドレスではない）を用いてCoAPディスカバリを活用する
- 広く知られているURI /oic/res対してマルチキャストで RETRIEVE (CoAP GET) を送信
- レスポンスはリンクの配列であり、それぞれのリンクは応答サーバにホストされたリソースを表す
- リンクから取得可能な情報：
  - href
  - 関係性（self link, hosted link, bridged link）
  - エンドポイント・バインド
  - サポートされているインターフェイス
  - リソースの可観測性

エンコーディング・スキーム： CBOR

- OCFにおいては、全てのものがリソースとなる
- 全てのリソースはJSONスキーマで指定され、関連するAPIはRAMLで定義される
- OCFでは、CBORをデフォルトの転送用エンコーディング方式に採用

	CBOR	*JSON*	*XML/EXI*
説明	**-JSONデータモデルに基づく簡潔なバイナリオブジェクト表現**	- 軽量な、テキストベースの、特定の言語に依拠しないデータ交換フォーマット	- XML用のバイナリ形式による圧縮の標準
標準	**IETF RFC 7049**	*IETF RFC 7159*	*W3C Efficient XML Interchange Format 1.0*
コンテンツの種類	**/application/vnd.ocf+cbor**	*/application/json*	*/application/exi*
OCF M/O	**必須**	サポート可能	サポート可能

必要に応じて将来的なバージョンで検討

コレクション・リソース

- OCFコレクション：他のOCFリソースに対するひとつ以上のレファレンス（OCFリンクとして定められる）を有するOCFリソース
- OCFリンクはRFC 5988によって定められているタイプされた「ウェブリンク」を取り入れ、また拡張する
- コレクションのわかりやすい例としては、/oic/res （ディスカバリ・リソース）が挙げられる
  - リソースモデルにおける一部のリソースはコレクションである

イントロスペクション

- 「なぜ」
  - 既存のAllJoynフレームワークと同等である
- 「なにを」
  - デバイスの説明はネット上で入手可能
  - デバイスの説明：
    - 全てのエンドポイントを列挙
    - エンドポイント毎に
      - どの手法が採用されるのか
        - » 手法毎のクエリパラメータ
        - » ペイロードの定義（リクエストとレスポンス）
- 「どのように」
  - RAMLおよびJSONにおいて説明されているデータを、CBORでエンコードされたSwagger2.0ドキュメントとして転送
    - ペイロードをJSONレベルで説明
      - プロパティ名
      - 種類
      - レンジ

イントロスペクション：目標

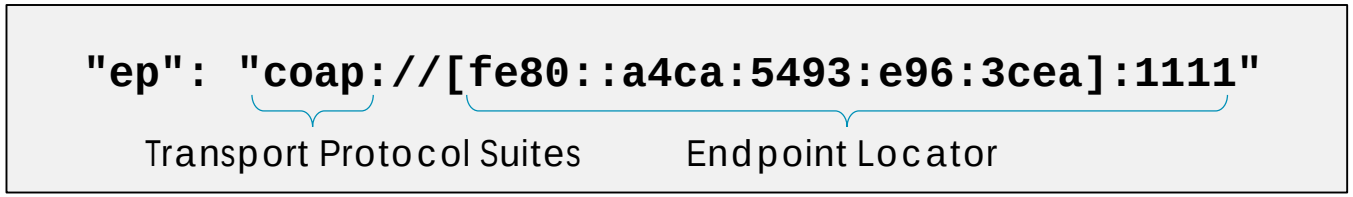
- 現状の仕事のやり方を損なわない：例として、RAML＋JSONはそのままの形で、転送されるSwaggerの定義のインプットに使用する
- 既に調査済みのものと変わらない制約条件を維持し、
  - 転送されるファイルはひとつ。例として、定義は以下のものを 含むべきである
    - 全てのエンドポイント、メソッド、クエリパラメータ、ペイロードの定義
  - ファイルのダウンロードに必要なネゴシエーションも現状と同じ種類のものとする
- 但し、今回はSwagger2.0ファイルである点だけが異なる

エンドポイントの概要

- 定義
  - （OCF）エンドポイントは、特定のトランスポートプロトコル群（例： CoAP over UDP over IPv6）におけるリクエスト/レスポンス メッセージの発信元もしくは送信先と定義される。エンドポイントの詳細な定義は、使用されるトランスポートプロトコル群によって異なる
    - 例：CoAP/UDP/IPv6においては、エンドポイントは「IPアドレス+ポート番号」となる
- OCFデバイスにおけるエンドポイントの特徴
  - それぞれのOCFデバイスは、リクエスト/レスポンス メッセージをやりとりできる少なくとも一つのエンドポイントに関連付けられる必要がある
    - メッセージがエンドポイントへと送信された場合、このメッセージはエンドポイントに関連付けられたOCFデバイスまで届かなければならない。エンドポイントにリクエストメッセージが届けられた際は、パスコンポーネントのみでターゲットリソースの位置が特定できる
  - OCFデバイスは、複数のエンドポイントに関連付けることが可能である
    - 例：OCFデバイスはCoAP およびHTTPの双方をサポートできる
  - リクエストURIでターゲットリソースを明確化できる場合のみ、ひとつのエンドポイントを複数のOCFデバイスで共有することができる

エンドポイント情報

- エンドポイント情報
  - エンドポイントは、下記のエンドポイント情報で識別される：
    - （1）トランスポートプロトコル群＋エンドポイントロケータは **ep**、（2）優先順位は **pri**



  - “ep”の現状のリストおよび、それぞれに対応するトランスポートプロトコル群

トランスポートプロトコル群	スキーム	エンドポイントロケータ	“ep” 値の例
coap + udp + ip	coap	IP address + port number	coap://[fe80::b1d6]:1111
coaps + udp + ip	coaps	IP address + port number	coaps://[fe80::b1d6]:1122
coap + tcp + ip	coap+tcp	IP address + port number	coap+tcp://[2001:db8:a::123]:2222
coaps + tcp + ip	coaps+tcp	IP address + port number	coaps+tcp://[2001:db8:a::123]:2233
http + tcp + ip	http	IP address + port number	http://[2001:db8:a::123]:1111
https + tcp + ip	https	IP address + port number	https://[2001:db8:a::123]:1122

エンドポイント情報用のepsパラメータ

- リンクにエンドポイント情報を埋め込むための新規パラメータ "eps"
  - “eps” はその値として複数の項目を有し、それぞれの項目はふたつの重要な値のペア（“ep” および “pri“）によって構成される。”ep”は必須である一方、 “pri”はオプショナルである

```
{
  "anchor": "ocf://light_device_id",
  "href": "/myLightSwitch",
  "rt": ["oic.r.switch.binary"],
  "if": ["oic.if.a", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coap://[fe80::b1d6]:1111", "pri": 2}, {"ep": "coaps://[fe80::b1d6]:1122"}]
}
```

  - “anchor” はホストとなるOCFデバイス、 “href”はターゲットリソース、そして“eps” はターゲットリソースのためのふたつのエンドポイントをそれぞれ示す
  - リンクのターゲットリソースがセキュアな接続（例：CoAPS）を要求する場合は、必要な情報（例：ポート番号）は“eps”のパラメータで示されるべきである

/oic/resにおける“eps”パラメータを使用したエンドポイント情報

/oic/res


```
[
 { "href": "/oic/res",
   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989/oic/res",
   "rel": "self",
   "rt": ["oic.wk.res"],
   "if": ["oic.if.ll", "oic.if.baseline"],
   "p": {"bm": 3},
   "eps": [{"ep": "coaps://[fe80::b1d6]:44444"}] },
 { "href": "/oic/p",
   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
   "rt": ["oic.wk.p"],
   "if": ["oic.if.r", "oic.if.baseline"],
   "p": {"bm": 3},
   "eps": [{"ep": "coap://[fe80::b1d6]:44444"}, {"ep": "coaps://[fe80::b1d6]:11111"} ] },
 { "href": "/oic/d",
   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
   "rt": ["oic.wk.d", "oic.d.light"],
   "if": ["oic.if.r", "oic.if.baseline"],
   "p": {"bm": 3},
   "eps": [{"ep": "coap://[fe80::b1d6]:44444"}, {"ep": "coaps://[fe80::b1d6]:11111"} ] },
 { "href": "/myLight",
   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
   "rt": ["oic.r.switch.binary"],
   "if": ["oic.if.a", "oic.if.baseline"],
   "p": {"bm": 3},
   "eps": [{"ep": "coap://[fe80::b1d6]:44444"}, {"ep": "coaps://[fe80::b1d6]:11111"} ] }
]
```


それぞれの
ターゲットリソースの
エンドポイント情報

バージョニング

ペイロードのバージョニング

- **目的：**クライアントとサーバがそれぞれのペイロードを理解できる
- **手法：**CoAPヘッダー内のエンコーディング情報およびリソースモデル

デバイスレベルのバージョニング

- **目的：**OCFデバイスがお互いのバージョン情報を把握できる
- **手法：**/oic/dリソース内のdmv（データモデルバージョン）およびicv （仕様バージョン）

ペイロードのバージョニング

Media Type	ID
application/cbor	60
application/vnd.ocf+cbor	10000

コンテンツフォーマット

CoAP Option Number	Name	Format	Length (bytes)
2049	Accept Version	uint	2
2053	Content-Format Version	uint	2

オプション番号

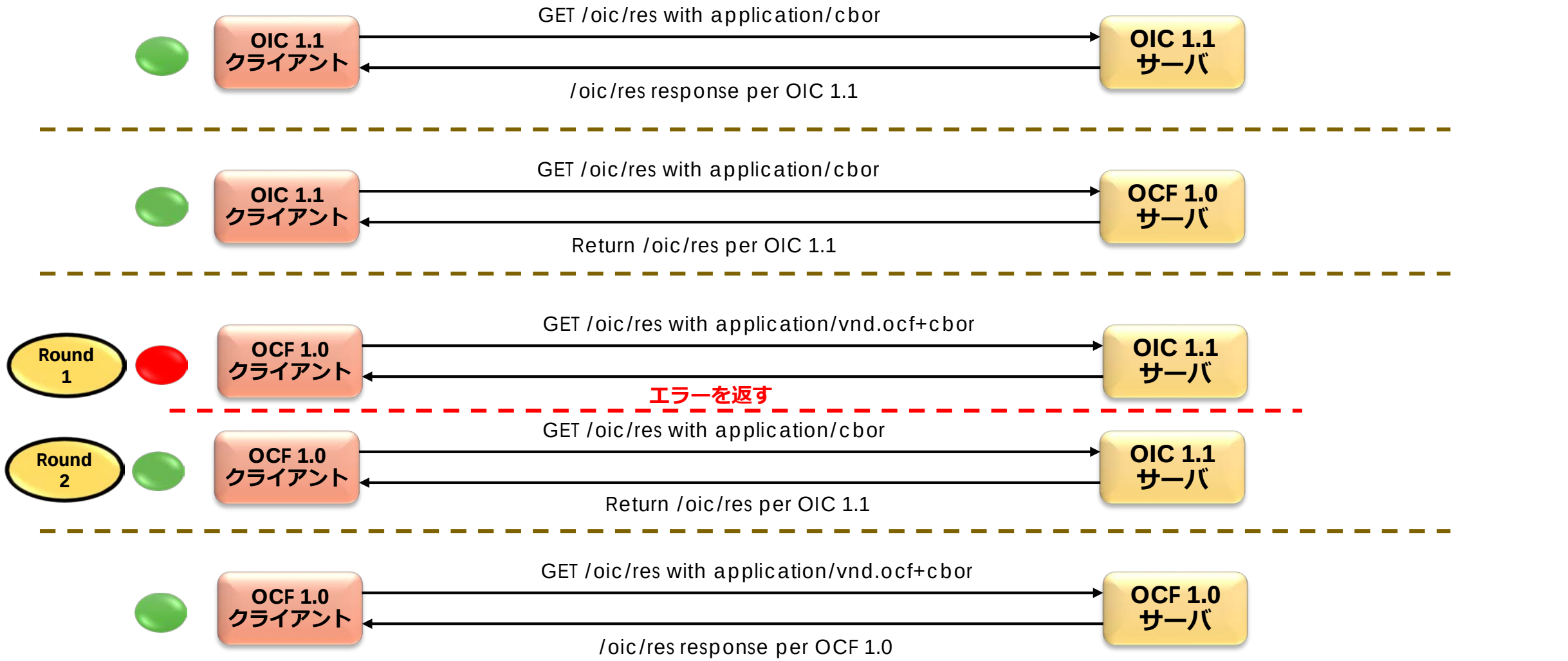
バージョン表現

	Major Version								Minor Version		Sub Version					
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

バージョンの例

OCF version	Binary representation	Integer value
1.0.0	0000 1000 0000 0000	2048
1.1.0	0000 1000 0100 0000	2112

ペイロードバージョニングのユースケースおよびポリシー



Block Transfer with CoAP Messaging

- 軽量かつ制限のあるIoTデバイスから予想される小さなペイロードに関しては、基本的なCoAPメッセージが十分に機能する
- 将来的にアプリケーションは、より大きなペイロードを扱うことが想定される
- CoAPデータグラムの容量を超えるコンテンツペイロードの リトリーブリクエストを受けた全てのOCFサーバは、IETF RFC 7959に定義されるCoAP blockwise transferを使用するものとする

インフラストラクチャの接続性

- 現在のスコープは、プロキシマルネットワークに重点を置いた取り組みとなっている
- 進行中のプロジェクト活動は、OCFによって生み出された機能性をより広範囲なエリア（クラウド等、LANを超えるもの）における接続性の実現に転用
  - ネイティブな CoAP
  - リソースディレクトリ
  - リソースホスト
  - 既にサポートされている Observe パターンに Pub-Sub パターンを追加

OCF コンポーネントの定義付け (on top of CORE)

- OCFサーバ
  - *device identifier*により定義される：規格化されたデバイス名
  - デバイス毎の必須OCFリソースタイプのリスト
  - OCFクライアントは暗黙的にOCFサーバの「逆側」と定められている点を踏まえる必要がある
    - 現在、OCFは交信シーケンスを要求していない
    - リソースタイプにおける全てのインスタンスは、いつでもいかなるOCFクライアントとも（いずれの方向にも）トークすることが許可されている
- OCFリソースタイプ
  - *resource identifier*により定義される：規格化されたリソース名
  - リソースタイプ毎の必須プロパティのリスト
  - リソースタイプ毎に許可されたアクション（read/readwrite/その他）のリスト
  - 全てのOCFリソースタイプIDはIANAに登録されている：
    http://www.iana.org/assignments/core-parameters/core-parameters.xhtml

ベンダーによる機能拡張

- ベンダーは以下の権限を有する：
  - 自らの定義による（OCFで標準化されていない） リソースタイプの作成
  - 自らの定義による（OCFで標準化されていない） デバイスタイプの作成
  - 既存のデバイスを追加的な（定められていない）リソースタイプで拡張
    - 標準化されたリソースタイプを使用することができる
    - ベンダーによって定義されたリソースタイプも使用することができる
- ベンダーによる全ての機能拡張はOCFによって定義された命名規則に準拠する

インフラストラクチャー：セキュリティの仕様

概要

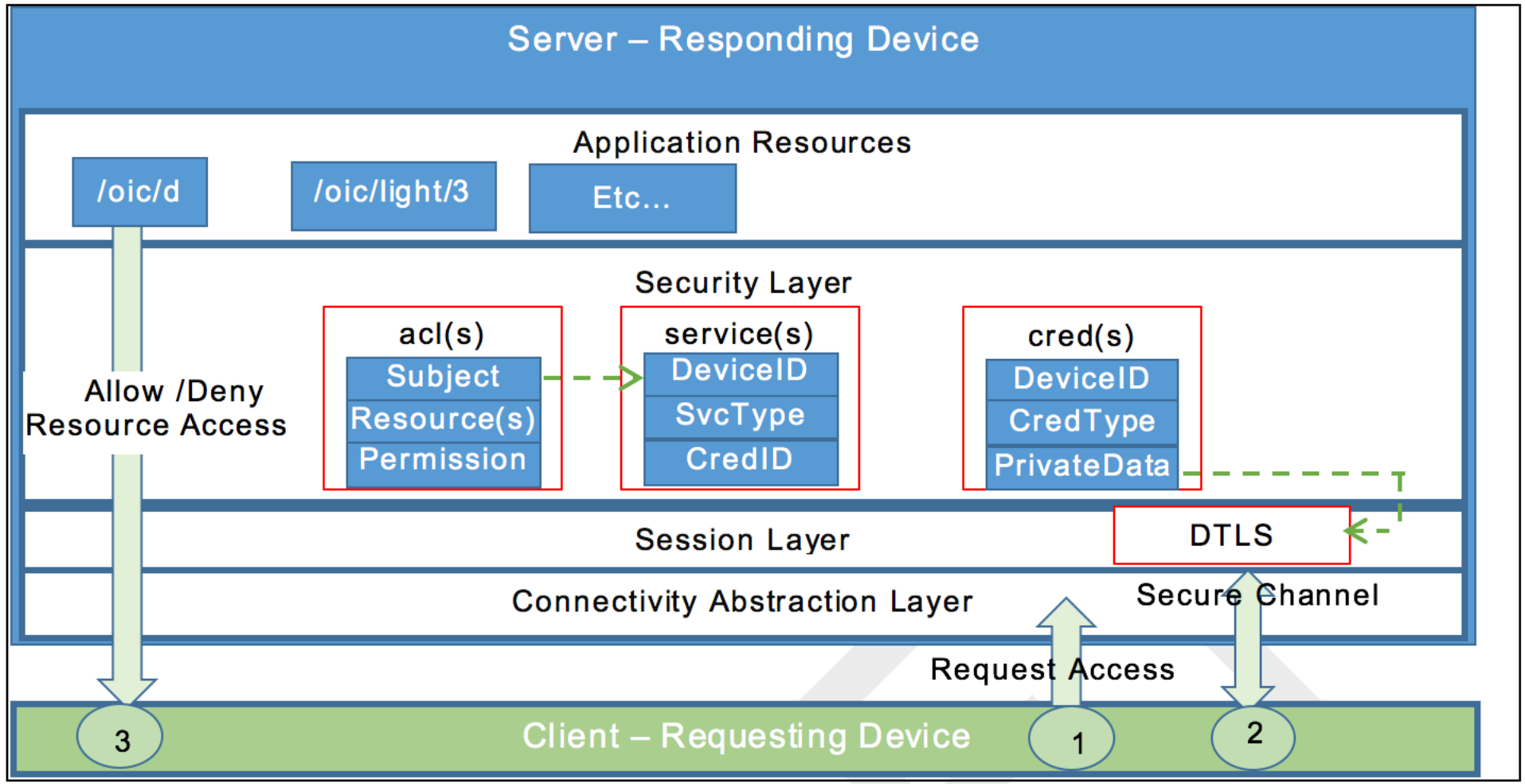
OCF セキュリティ：概要

- OCFが特に留意する点としては、以下のものが挙げられる：
  - **デバイスのアイデンティティ**（変更不能で、重複せず、認証可能であること）
  - **オンボーディング**（**認証、承認、監査（AAA）**を含む）
  - **機密性**（データ及び通信内容を保護）
  - **完全性**（リソース、デバイスの状態、およびトランジションは全て管理されている）
  - **可用性**（デバイスレベルのみならず、稼働するネットワークにも影響を与えないようセキュリティが付与されている）
  - **ライフサイクルマネジメント**（セキュアなソフトウェアアップデートおよび認証メカニズムを含む）
  - **将来的なセキュリティ**（証明書の種類やアルゴリズムを検討し、OFCデバイスに関連するセキュリティ分野における現在と将来の変化に対応）
- OCF の暗号化鍵管理は、デバイスの保護と認証をサポートしている
- OCFは、認証管理にアクセス制御リスト（ACLs） を使用している
- セキュアなデバイス所有権の移動が、ネットワークにデバイスを追加する際のアタックを防ぐことに寄与する

セキュリティの原則

- **リソース：**デバイスの種類、データ、およびインターフェイスを定義するデータ構造。それぞれ生成・参照・更新・削除が可能であり、適切なアクセス制御に基づき通知を設定することができる
- **アクセス制御エントリ（ACEs）およびアクセス制御リスト（ACLs）**はそれぞれ、ひとつのデバイスにリソースへのアクセス権を与える許可のエントリとコレクションである
- **アクセス管理者サービス（AMS）**は、アクセス制御の許可を生成・認証する
- **証明書管理サービス（CMS）**は、セキュリティ証明書を発行・管理する許可を得たデバイスの名称とリソースタイプである
- **セキュリティ・バーチャル・リソース（SVRs）**は、許可やアクセス管理が厳しく制限された、特別なセキュリティリソースである
- **オンボーディングツール（OBT）**は、OCFデバイスをローカルネットワークに組み込む作業をサポートする、信頼されたプラットフォームである

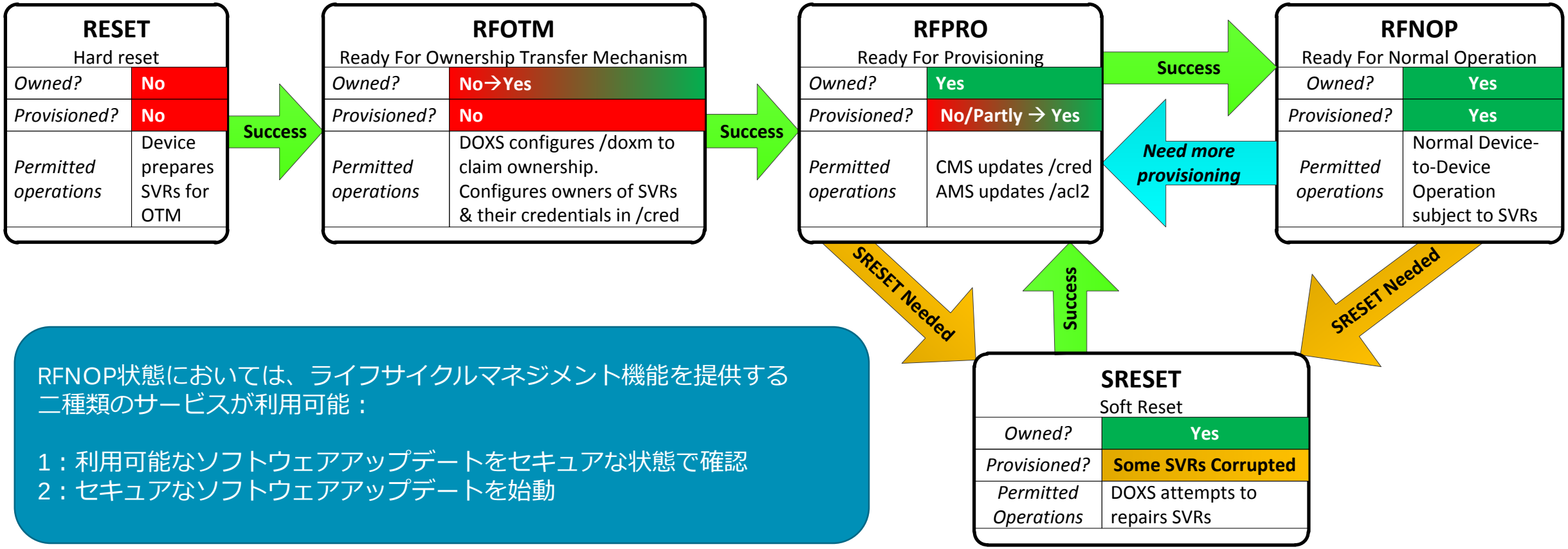
OCFセキュリティはどのようにデバイス・リソースを守るのか？



簡素化されたオンボーディング・シーケンス

- *Unowned*（オーナーシップがない状態の）デバイスのブート
- **ディスカバリ（セキュリティが確保されていない状態）**
  - DOXSがUnownedデバイスをディスカバリするためのマルチキャストを送信 — no TLS
  - Unownedデバイスが、サポートされているOTMのリストを含めてリプライ — no TLS
- **オーナーシップの移動**
  - DOXSがOTMを選択し、新しいデバイス向けに設定 — no TLS
  - DOXSおよびUnownedデバイスが、TLSハンドシェークを含むOTMを実行 — TLS
  - DOXSが、SVRが自身ならびにCMS、AMSを承認するよう設定 — TLS
  - これにより、デバイスのオーナーシップがある状態となる
- **プロビジョニング：**
  - CMSが証明書を確保し、AMSはアクセスポリシーを確保 — TLS
  - デバイスのプロビジョニングが完了し、通常オペレーションの開始が可能となる
- **通常オペレーション** — TLS or no TLS
  - プロビジョニングのプロセスに戻ることにより、証明書やアクセスポリシーのアップデートが可能

デバイスのプロビジョニングにおける各状態



RFNOP状態においては、ライフサイクルマネジメント機能を提供する
二種類のサービスが利用可能：

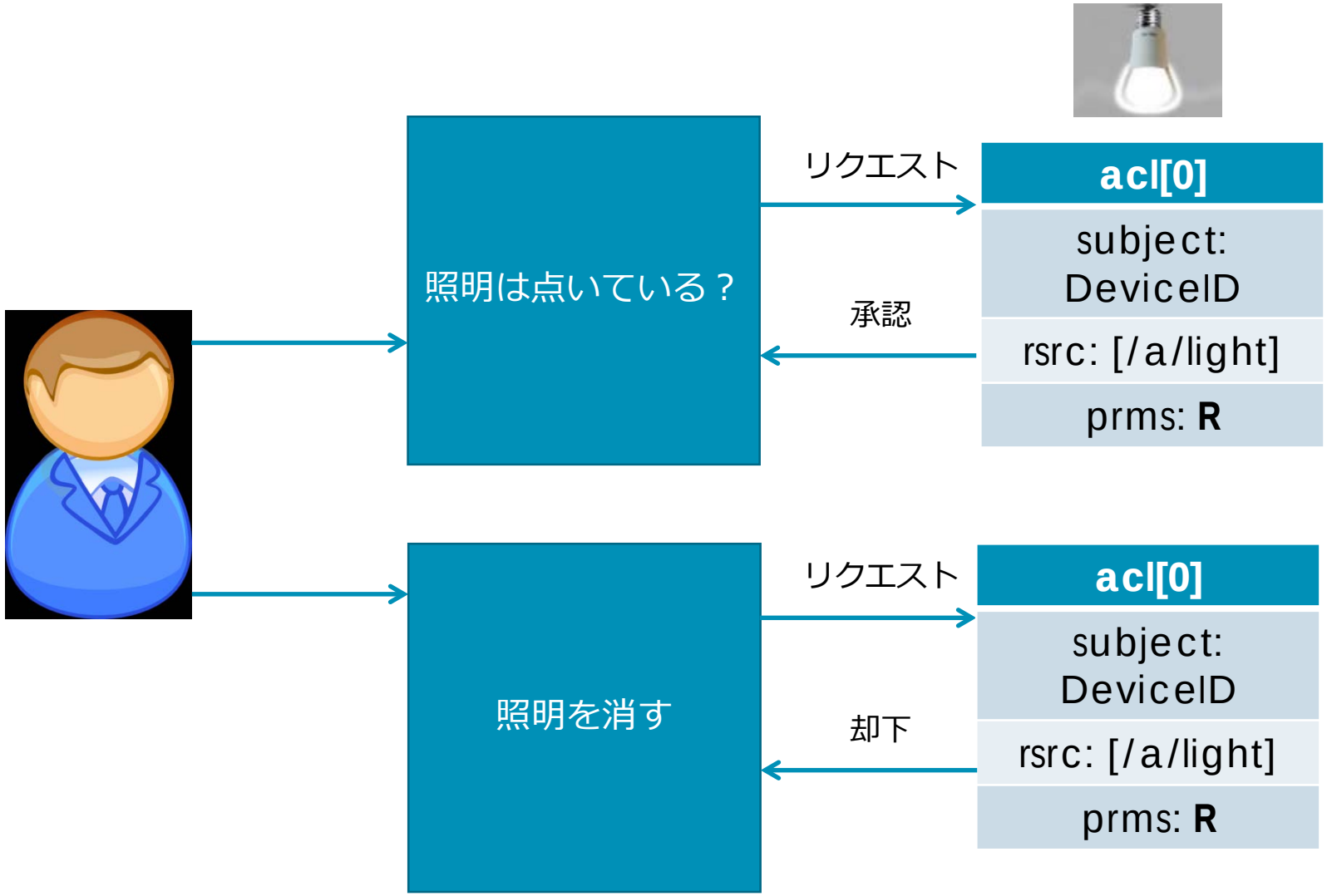
1：利用可能なソフトウェアアップデートをセキュアな状態で確認
2：セキュアなソフトウェアアップデートを始動

図示されていないが、デバイスはどの状態からも*RESET*状態に移行可能

証明書の管理

- OCFデバイスは、セキュアな通信を確保する際、対称・非対称の双方の証明書の使用をサポートできる
  - 対称キーは必須
  - 証明書ならびに公開鍵・非公開鍵がサポートされている
- 不足している証明書はCMSから取得可能
- 証明書には、期限が設定されている場合がある
  - 期限が切れた証明書は更新できる

アクセスコントロール



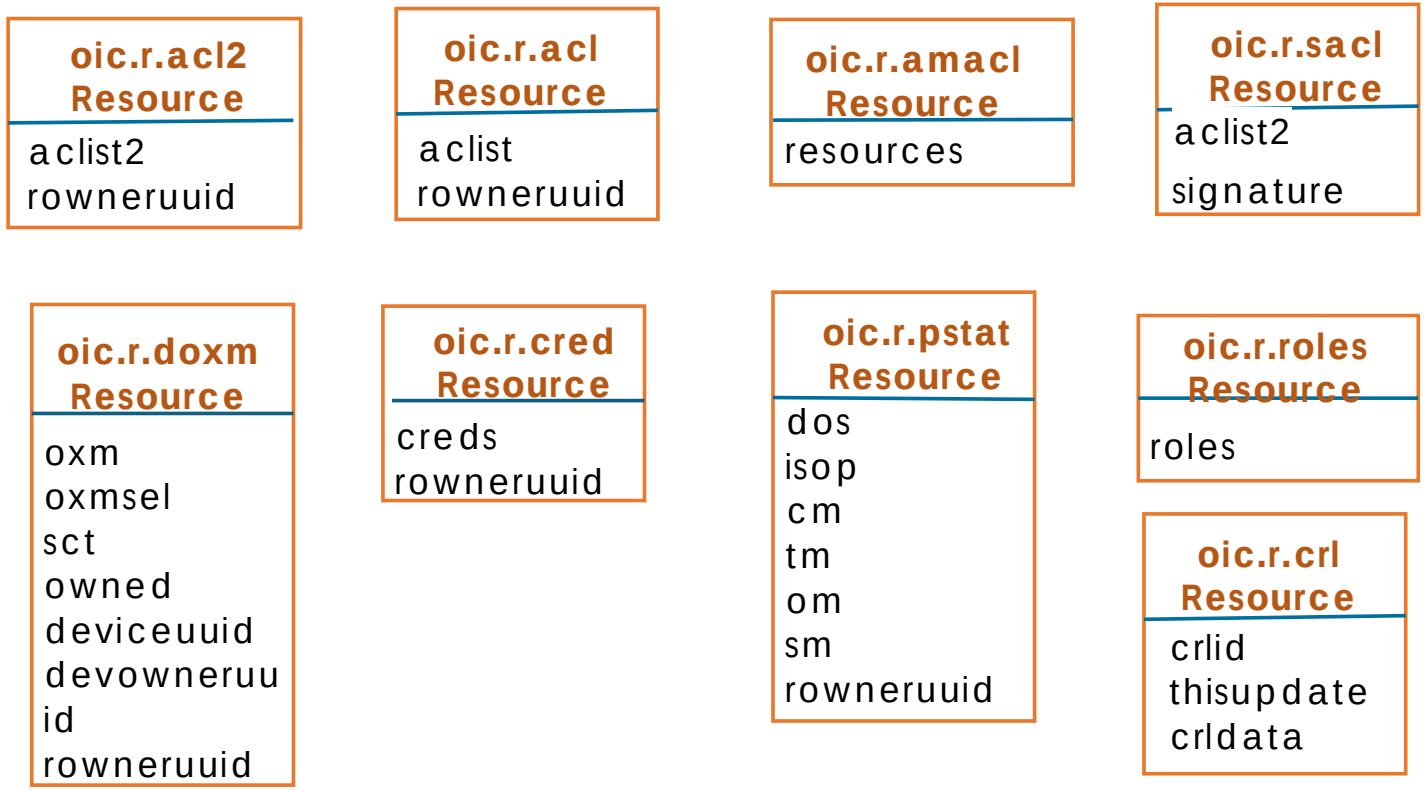
アクセスコントロール

- アクセスを求めるエンティティのCRUDNアクセスを制御するために、OCFサーバのリソースを守る
  - OCFサーバに対するリクエストは、全てACL（アクセス制御リスト）ポリシーのチェックを受ける
  - OCFサーバーがホストするリソースにはACE（アクセス制御エントリ）ポリシーが適用される
  - それぞれのACEは、read もしくは write のオペレーションを許可する権限を有する
- 二種類のアクセス制御メカニズムをサポート：
  - サブジェクトベースアクセス制御（SBAC）
    - ACEが、リクエストする側のアイデンティティを特定する
  - ロールベースアクセス制御（RBAC）
    - ACEが、リクエストを行っているエンティティにおける、許可される役割（ロール）を特定する
- ACLは、AMSを通じて変更・アップデートが可能
- ACLポリシーは、OCFサーバ側のみに適用される

セキュリティ・バーチャル・リソース (SVR)

- OCFの定義では、SVRs（セキュリティ・バーチャル・リソース）はOCFセキュリティ関連の機能を実行するデバイスとしている
- デバイスオーナーシップ移動リソース（/oic/sec/doxm）はデバイスのオーナーシップ状態を管理
- プロビジョニングリソース（/oic/sec/pstat）はデバイスのプロビジョニング状態を管理
- 証明書リソース（/oic/sec/cred）はデバイスの証明書を管理
  - 証明書リソースはセキュアな通信を確保するために使用される
  - 証明書取消リストリソース（/oic/sec/crl）は証明書の取り消しを管理
  - ロールリソース（/oic/sec/roles）は役割に対応した証明書を管理
  - 証明書署名リクエストリソース（/oic/sec/csr）は、DOXSにより証明書に署名を行うために使用される
  - セキュリティのハードニングは /oic/sec/cred リソースに該当する
- アクセス制御リスト（/oic/sec/acl）はリソースサーバのアクセス制御リストを管理する
  - アクセス管理者ACLリソース（/oic/sec/amacl）はACLを強制する際のAMSを選定する
  - 署名済みACLリソース（/oic/sec/sacl）はACLポリシーの署名に使用される

セキュリティ・バーチャル・リソース (SVR)



メッセージの完全性および機密性

- クライアントとサーバの間のセキュアな通信は、盗聴・改竄・再送攻撃から守られている
- ユニキャストのメッセージはDTLSもしくはTLSでセキュアされる。一方、マルチキャストのメッセージはセキュアされていない
- 全てのセキュアな通信は、署名・暗号化されている
- 通信を行うデバイスは、お互いを認証することが求められ、使用可能な証明書がなければ通信を行うことができない。証明書が不足している場合は、CMSを通して取得が可能となっている
- 送信側デバイスは、選択された暗号スイートの定義に基づきメッセージを暗号化ならびに認証し、受信側デバイスはメッセージを検証・復号する
- セキュアなユニキャストメッセージは、デバイスのオーナーシップ移動ならびに通常オペレーションの際には指定された暗号スイートを使用する（対称キーと非対称証明書に関して）

インフラストラクチャー：ブリッジングの仕様

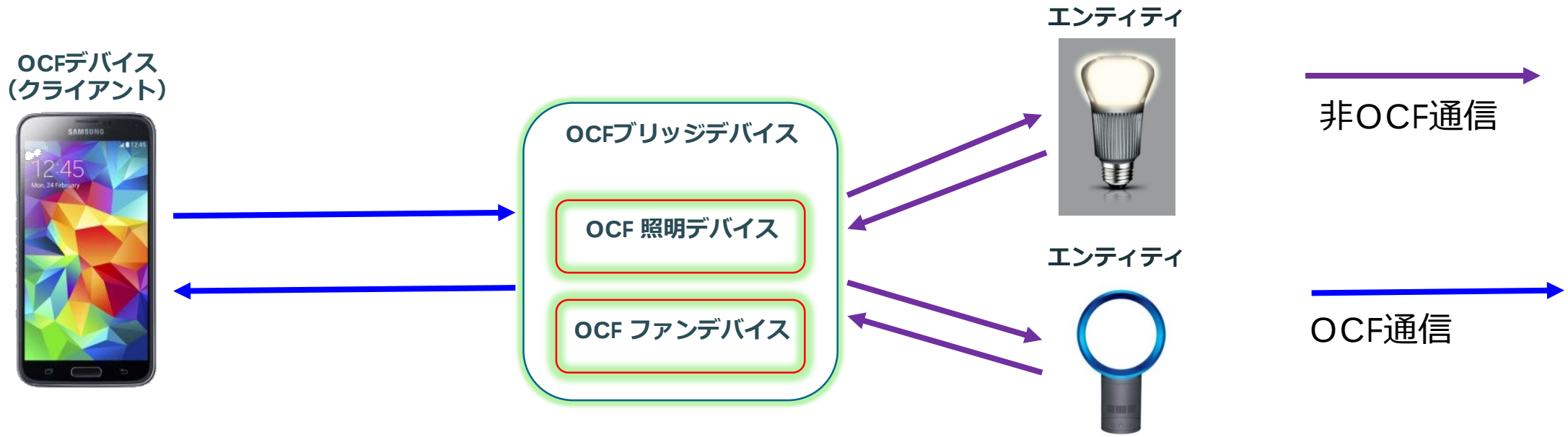
概要

ブリッジングの仕様

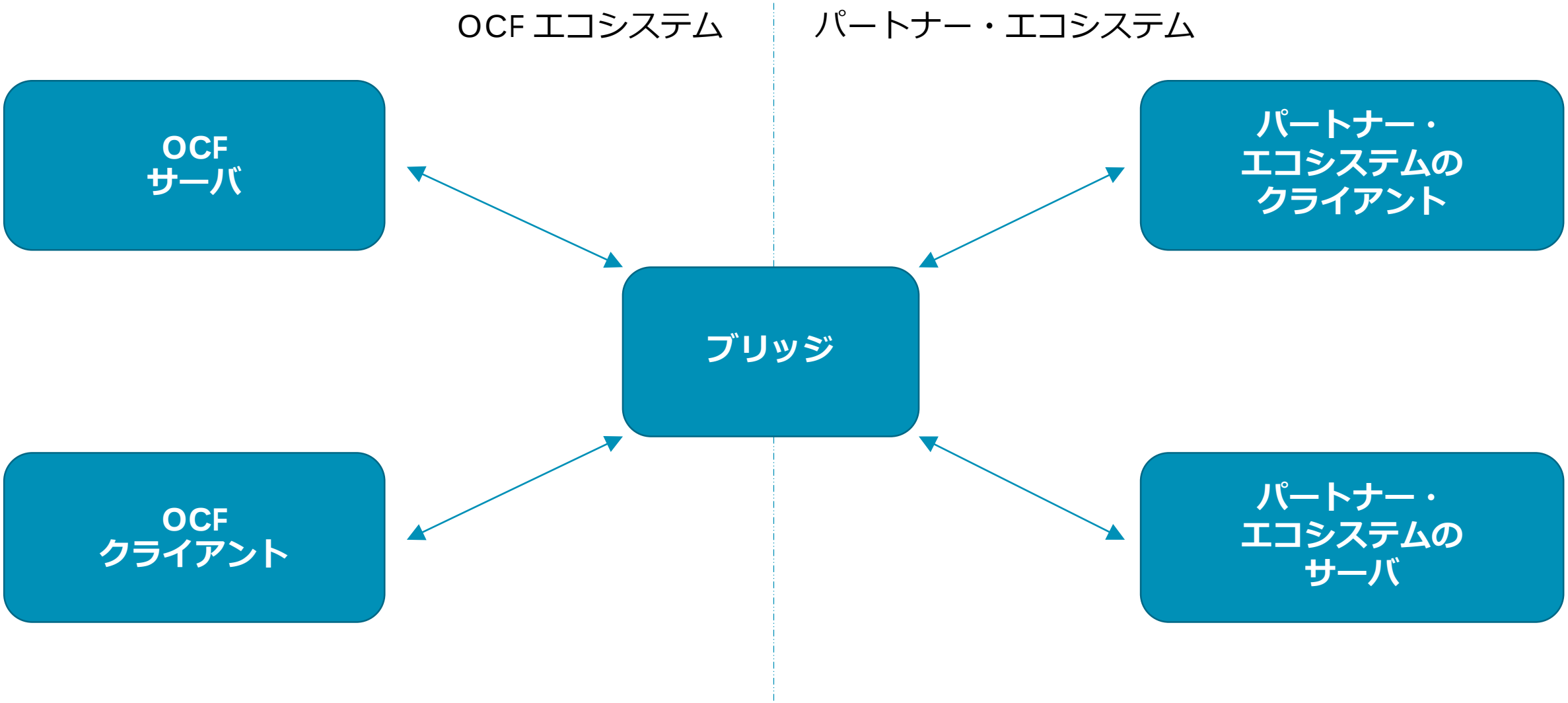
- OCFおよび非OCFエコシステムにおけるデバイス間の双方向変換フレームワークを定める
- OCFと非OCFエコシステムの間の変換における一般的な要件を定める
  - リソースディスカバリ、メッセージ変換、セキュリティ、ならびに複数のブリッジの取り扱いにおける要件
- OCFとAllJoynエコシステムの間の変換における具体的な要件を定める
  - コア・リソースのマッピング、誤差伝搬、およびカスタマイズされたリソースタイプのアルゴリズム的変換における要件
  - 一般的なリソースタイプの変換における、OCFからAllJoynへのマッピングの仕様を指す

OCFブリッジ：定義

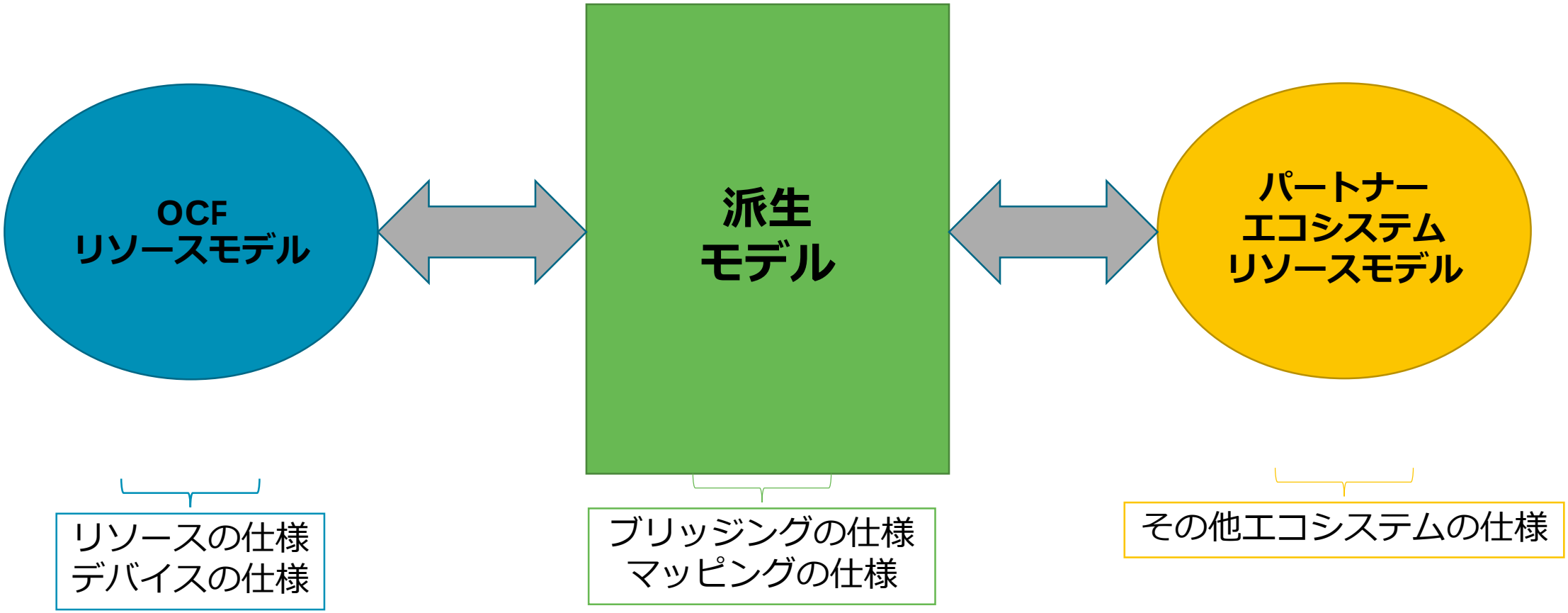
- OCFブリッジは、少なくともひとつの非OCFデバイス（ブリッジされたデバイス）の代理として、OCFネットワーク上でバーチャルOCFデバイスの役割を果たすデバイスである
- ブリッジされたデバイス事態は、OCFのスコープ外となる
- 「通常の」OCFデバイスとバーチャルOCFデバイスの唯一の違いは、後者がOCFブリッジデバイスに内包されているという点だけである
- OCFブリッジデバイスは、ネットワーク上では「oic.d.bridge」の「rt」として示される。そのようなデバイスが発見された際は、そのデバイスのディスカバラブルなリソースが、ブリッジされたデバイスの特徴を示す



ブリッジングの基本コンセプト：双方向オペレーション



ブリッジングの基本コンセプト：データモデル



ブリッジングにおけるセキュリティ

- OCFブリッジはメッセージペイロードを変換するため、信頼できるエンティティでなければならない
- OCFブリッジそのものに加えて、OCFブリッジを通じて開示されるバーチャルデバイスは、全てオンボード（オーナーシップの移動）される必要があり、セキュアなオペレーションのためにプロビジョニングされなければならない
- OCFブリッジによって開示される各バーチャルデバイスは、接続されるエコシステムのセキュリティ要件を満たす必要がある
- ブリッジングは、OCFブリッジ/OCFデバイス間、ならびにOCFブリッジ/ブリッジされたデバイス間の特定の通信を遮断するためのメカニズムを定めている。このような細やかな制御が、同程度のセキュリティ機能を有さない異なるエコシステム間のコミュニケーションを制御する能力をアドミニストレータにもたらしている

OPEN CONNECTIVITY FOUNDATION™

リソースモデル：リソースタイプの仕様

概要

リソースの仕様

- OCFデバイスに使用されている、再利用可能なリソースのリスト
  - OCF1.0の時点では合計74のリソースタイプが定義されている
  - コア定義を使用
- それぞれのリソースの定義は以下の要素を含む：
  - 重複していない識別子（identifier）：（rt）
  - デフォルトインターフェイスや、その他のサポートされているインターフェイスの認識
  - サポートされているメソッドのリスト
  - サポートされているペイロードを定義するJSONスキーマの、メソッド毎のリスト
  - リソースが開示することになるプロパティの詳細なリスト

リソースは RESTful API Modelling Language (RAML) および *Swagger2.0*によって定められる

定義されたリソースタイプのサンプルセット OIC 1.1（1/2）

リソースの種類	用途
Air Flow	Indoor Environment Control
Air Flow Control	
Battery	Device Control
Binary switch	Device Control
Brightness	Lighting Control
Colour Chroma	
Colour RGB	
Dimming	
Door	Indoor Environment Control
Energy Consumption	Energy Management
Energy Usage	
Humidity	Indoor Environment Control
Icemaker	Device Control

リソースの種類	用途
Lock	Keyless Entry
Lock Code	
Mode	Device Control
Open Level	
Operational State	
Ramp Time	Lighting Control
Refrigeration	Device Control
Temperature	Indoor Environment Control
Time Period	Device Control

定義されたリソースタイプのサンプルセット OIC 1.1（2/2）

リソースの種類	用途
Audio	TV, Home Entertainment
Auto Focus	IP Camera
Auto White Balance	IP Camera
Automatic Document Feeder	Scanner Support
Button	Device Control
Colour Saturation	IP Camera
DRLC	Smart Energy
Energy Overload	Smart Energy
Media	IP Camera
Media Source List	TV, Home Entertainment
Movement (Linear)	Robot Cleaner
Night Mode	IP Camera
PTZ	IP Camera
Signal Strength	Proximity

リソースの種類	用途
Acceleration	Extended Sensor Set (for a Generic Sensor Device)
Activity Count	
Atmospheric Pressure	
Carbon Dioxide	
Carbon Monoxide	
Contact	
Glass Break	
Heart Rate Zone	
Illuminance	
Magnetic Field Direction	
Presence	
Radiation (UV)	
Sleep	
Smoke	
Three Axis	
Touch	
Water	

OCFの定義されたリソースタイプの完全なリストはこちら： https://oneiota.org

新規リソースタイプ： OCF 1.0

リソースの種類	用途
Air Quality	Indoor Environment Control
Air Quality Collection	Indoor Environment Control
Consumable	Device Control
Consumable Collection	Device Control
Delay Defrost	Energy Star
Ecomode	Device Control
Heating Zone	Device Control
Heating Zone Collection	Device Control
Selectable Levels	Device Control
Value Conditional	Notifications

リソースタイプは特定の条件下で必須となる。もしOCFサーバが既知のOCFリソースをホストした場合、
リソース定義に定められている、そのリソースに該当する 全ての基準要件を満たす必要がある。

OPEN CONNECTIVITY FOUNDATION™

リソースモデル：
派生モデリング – OCF to AllJoyn マッピング

概要

概要

- OCFとAllJoynの間の相互作用をモデリングする
- 下記の、OCF白書において定義されている派生モデルシンタックスを使用（若干の小さな変更あり）：https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-Between-IoT-Ecosystems_v2-examples.pdf
- OCFが上位モデルであるという前提に基づいている。つまり、OCFから欠けていた全てのデバイスタイプやリソースタイプ（AllJoynインターフェイスの同等物としてのもの）は、同等のOCF仕様 において定義されている
- 以下の要素でマッピングを定義している：
  - デバイスタイプの等価性
  - リソースとインターフェイスの等価性
  - インターフェイス毎の、詳細なプロパティ別のマッピング（派生モデル）

派生モデルシンタックス

- 派生モデルは、一般的なJSONスキーマシンタックスを使用している。本質的には、派生モデルはOCFとAllJoynのデータモデル間のコンバージョンマッピングを提供するものである

```
"asa.environment.targethumidity": {
    "type": "object",
    "properties": {
      "targetvalue": {
        "type": "number",
        "description": "Measured value",
        "x-ocf-conversion": {
          "x-ocf-alias": "oic.r.humidity,oic.r.selectablelevels",
          "x-to-ocf": [
            "if minvalue != maxvalue, ocf.desiredhumidity = targetvalue;ocf.targetlevel = selectablehumiditylevels[0].",
            "if minvalue == maxvalue, ocf.targetlevel = targetvalue."
          ],
          "x-from-ocf": [
            "if x-ocf-alias == oic.r.humidity, targetvalue = desiredhumidity.",
            "if x-ocf-alias == oic.r.selectablelevels, targetvalue = targetlevel."
          ]
        }
      }
    }
```

AllJoyn インターフェイス名

AllJoyn プロパティ名

OCF 同等物リソースタイプ

“To OCF” ブロック

“From OCF” ブロック

デバイスタイプの等価性

分類	ASAデバイスタイプ	OCFデバイスタイプ	OCFデバイスタイプID
Air Care	Air Conditioner	Air Conditioner	oic.d.airconditioner
	AirPurifier	Air Purifier	oic.d.airpurifier
	AirQualityMonitor	Air Quality Monitor	oic.d.aqm
	Dehumidifier	Dehumidifier	oic.d.dehumidifier
	Humidifier	Humidifier	oic.d.humidifier
	ElectricFan	Fan	oic.d.fan
	Thermostat	Thermostat	oic.d.thermostat
Fabric Care	Clothes Washer	Washer	oic.d.washer
	Clothers Dryer	Dryer	oic.d.dryer
	Clothers Washer-Dryer	Washer-Dryer	oic.d.washerdryer
Food Preservation	Refrigerator	Refrigerator	oic.d.refrigerator
	Ice Maker	Ice Maker (Resource)	oic.r.icemaker
	Freezer	Freezer	oic.d.freezer
Food Preparation	Oven	Oven	oic.d.oven
	Cooktop	Cooktop	oic.d.cooktop
	Cookerhood	Cooker Hood	oic.d.cookerhood
	Foodprobe	Food Probe	oic.d.foodprobe
Dish Care	Dishwasher	Dishwasher	oic.d.dishwasher
Floor Care	Robot Cleaner	Robot Cleaner	oic.d.robotcleaner
Entertainment	TV	Television	oic.d.tv
	Set Top box (STB)	Set Top Box	oic.d.stb

- 等価性をサポートするために追加されたデバイスタイプは、黄色でハイライトされている

インターフェイスからリソースへのマッピング

AllJoyn インターフェイス	OCF リソースタイプ名	OCF リソースタイプID	OCF インターフェイス
Environment.CurrentAirQuality	Air Quality Collection	oic.r.airqualitycollection	oic.if.s
Environment.CurrentAirQualityLevel	Air Quality Collection	oic.r.airqualitycollection	oic.if.s
Environment.CurrentHumidity	Humidity	oic.r.humidity	oic.if.s
Environment.CurrentTemperature	Temperature	oic.r.temperature	oic.if.s
Environment.TargetHumidity	Humidity	oic.r.humidity, oic.r.selectablelevels	oic.if.a
Environment.TargetTemperature	Temperature	oic.r.temperature	oic.if.a
Operation.AudioVolume	Audio Controls	oic.r.audio	oic.if.a
Operation.Channel	Not mapped		
Operation.ClimateControlMode	Mode	oic.r.mode	oic.if.a
	Operational State	oic.r.operational.state	oic.if.s
Operation.ClosedStatus	Door	oic.r.door	oic.if.s
Operation.CycleControl	Operational State	oic.r.operational.state	oic.if.s
Operation.FanSpeedLevel	Air Flow	oic.r.airflow	oic.if.a
Operation.HeatingZone	Heating Zone Collection	oic.r.heatingzonecollection	oic.if.s
Operation.HvacFanMode	Mode	oic.r.mode	oic.if.a
Operation.OnOffStatus	Binary Switch	oic.r.switch.binary	oic.if.s
Operation.OvenCyclePhase	Operational State	oic.r.operationalstate	oic.if.s

OPEN CONNECTIVITY FOUNDATION™

業種バーティカル別：スマートホームデバイスの仕様

概要

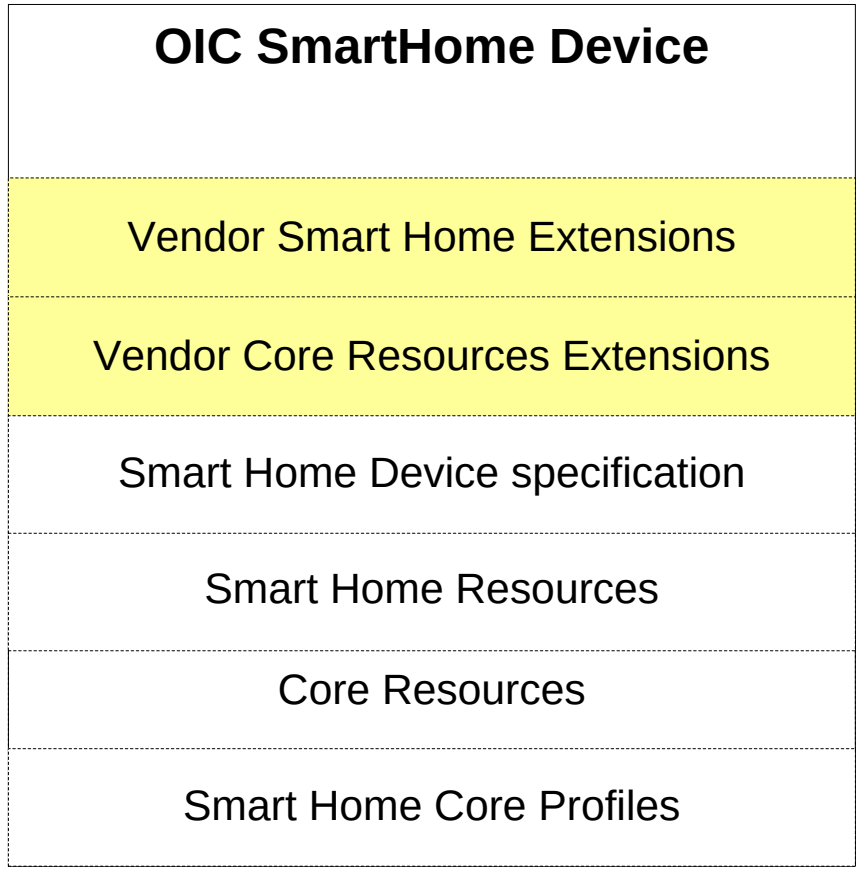
上位レイヤの仕様

- 仕様は二つの文書に分けられている：
  - デバイスの仕様（必要な場合は業種バーティカル別）
  - リソースの仕様（業種バーティカルに依存しない）

**デバイスの仕様は、リソースの仕様で定義されている
リソースを使用している**

デバイスの仕様

- 以下のプロファイルを含む：
  - コア仕様
  - セキュリティ仕様
- スマートホームデバイスのリストを含む
- 各スマートホームデバイスの定義は以下を含む：
  - 重複していない識別子（identifier）：（rt）
  - 必須リソースのリスト



OCFデバイスタイプの開示は必須である。もしOCFサーバが既知のOCFデバイスをホストした場合、
デバイス定義に定められている、そのデバイスに該当する 全ての基準要件を満たす必要がある。

スマートホームデバイスの種類 （1/2）

デバイスの種類	最低限のリソースセット
Air Conditioner	Binary Switch, Temperature
Air Purifier	Binary Switch
Air Quality Monitor	Air Quality Collection
Blind	Open Level
Camera	Media
Clothes Dryer	Binary Switch, Mode
Clothes Washer	Binary Switch, Mode
Clothes Washer/Dryer	Binary Switch, Operational State
Cooker Hood	Airflow Control, Binary Switch, Mode
Cooktop	Heating Zone Collection
Dehumidifier	Binary Switch, Humidity
Dishwasher	Binary Switch, Mode
Door	Open Level
Fan	Binary Switch

デバイスの種類	最低限のリソースセット
Food Probe	Temperature
Freezer	Temperature (2)
Garage Door	Door
Generic Sensor	Sensor
Humidifier	Binary Switch
Light	Binary Switch
Oven	Binary Switch, Temperature (2)
Printer	Binary Switch, Operational State
Printer (Multi-Function)	Binary Switch, Operational State (2), Automatic Document Feeder
Receiver	Binary Switch, Audio Media Source List (2)

スマートホームデバイスの種類（2/2）

デバイスの種類	最低限のリソースセット
Refrigerator	Binary Switch, Refrigeration, Temperature (2)
Robot Cleaner	Binary Switch, Mode
Scanner	Binary Switch, Operational State, Automatic Document Feeder
Security Panel	Mode
Set Top Box	Binary Switch
Smart Lock	Lock Status
Smart Plug	Binary Switch
Switch	Binary Switch
Television	Binary Switch, Audio, Media Source List
Thermostat	Temperature (2)
Water Valve	Open Level

Thank you!

- OCFの仕様にアクセス：
  https://openconnectivity.org/resources/specifications
- OCFへのお問合せ： admin@openconnectivity.org

OPEN CONNECTIVITY FOUNDATION™