

Operational Insights 1.0.0

[Base URL: dnspaces.io/api/am/v1]

Operational Insights (OI) は、資産、Internet of Things (IoT)、アラートシステム、運用ワークフローの管理、監視、最適化のための包括的な単一リソースです。テクノロジーに依存しないアプローチを使用するこのソリューションは、Wi-Fi、RFID、環境モニターなど、幅広いタグとセンサーを使用して、コネクテッドオペレーションを継続的に統合、モニター、および管理できます。

このドキュメントでは、次の Operational Insights (OI) API について説明します。

1. 新しいアセットの追加
2. フィルタを使用したアセットの取得
3. アセットのリストの取得
4. 単一のアセットの取得
5. アセットの更新
6. アセットの削除
7. 新しいカテゴリの追加
8. カテゴリのリストの取得
9. 単一のカテゴリの取得
10. カテゴリの更新
11. カテゴリの削除
12. 新しい部門の追加
13. 部門のリストの取得
14. 単一の部門の取得
15. 部門の更新
16. 部署の削除
17. 新しいカスタムフィールドの追加
18. カスタムフィールドのリストの取得
19. カスタムフィールドの更新
20. カスタムフィールドの削除
21. 新しいタグの追加
22. タグのリストの取得
23. 未割り当てのタグのリストの取得

- 24. 単一のタグの取得
- 25. タグの更新
- 26. タグの削除
- 27. アセットのロケーション履歴の取得
- 28. アセットのテレメトリ履歴の取得

これらの API にアクセスするには、認証用のアクセストークンが必要です。このトークンを取得するには、<https://dnaspaces.io/api/am/v1/auth/license/accesstoken> に GET リクエストを送信します。提供された永続的な API トークンを、プレフィックス「JWT」とともに、リクエストヘッダーのキー「Authorization」の下に添付します。この API は、キー「token」の下に API アクセストークンを見つけることができるオブジェクトを返します。この API アクセストークンを、プレフィックス「JWT」とともに、実行する API へのリクエストヘッダーのキー「Authorization」の下に添付します。トークンの有効期限は 30 分です。

スキーム

HTTPS

Authorize



Assets



POST

/entities/access/assets



この API は、アセットを作成するために使用されます。アセットを作成する前に、アセットに割り当てたいカテゴリ、部門、タグ、および/またはカスタムフィールドの ID があることを確認してください。各エンティティには、システムに追加されるときに ID が割り当てられます。

パラメータ

試してみる

名前

説明

名前

説明

body * 必須

オブジェクト
(本文)

新規アセット作成の要求本文：詳細は「NewAsset」スキーマを参照

値の例 | モデル

```
{
  "serial": "serial-123",
  "name": "name-123",
  "description": "Asset Description",
  "status": "Active",
  "tags": [
    1
  ],
  "category": 100,
  "subcategories": [
    "subcategory"
  ],
  "department": 200,
  "customData": {
    "1": "test"
  },
  "categoryCustomData": {
    "1": "test"
  },
  "site": {
    "id": "abc",
    "level": "FLOOR",
    "name": "campus -> building -> floor",
  }
}
```

パラメータコンテンツタイプ

application/json ▼

応答

応答コンテンツタイプ

application/json ▼

コード

説明

201

エンティティが正常に作成されました

値の例 | モデル

```
{
  "id": 0
}
```

コード 説明

400

要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

409

シリアル番号が一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/assets



この API は、アセットのリストを取得するために使用されます。この API では、クエリパラメータで制限とページを指定する必要があります。この API が現在サポートしている上限は 100 です。ページはページ 0 から始まります。次の一連のアセットを取得するには、ページの値を増分するだけです。

パラメータ

試してみる

名前 説明

limit * 必須

一度に取得できる数。100 が最大です。

整数

(クエリ)

limit : 一度に取得できる数。100 が最大です。

page * 必須

どのページから取得を開始するか。ページ 0 から始まります

整数

(クエリ)

page : どのページから取得を開始するか。ページ 0 から始まります

応答

応答コンテンツタイプ

application/json



コード

説明

200

すべてのアセットを正常に取得しました

値の例 | モデル

```
[
  {
    "id": 1,
    "serial": "serial-123",
    "name": "name-123",
    "location": {
      "x": 100,
      "y": 100,
      "floorId": "abc",
      "zones": [
        {
          "color": "#abc",
          "id": "abc",
          "name": "abc"
        }
      ],
      "chokePoints": [
        {
          "id": "abc",
          "name": "abc"
        }
      ],
      "hierarchy": "abc",
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

コード	説明
-----	----

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/assets/{id}



この API は、ID を使用して単一のアセットを取得するために使用されます。

パラメータ

試してみる

名前	説明
----	----

id * 必須

アセットの ID

整数

(パス)

id: アセットの ID

応答

応答コンテンツタイプ

application/json



コード	説明
-----	----

200

単一のアセットを正常に取得しました

値の例 | モデル

```
{
  "id": 1,
  "serial": "serial-123",
  "name": "name-123",
  "location": {
    "x": 100,
    "y": 100,
    "floorId": "abc",
    "zones": [
      {
        "color": "#abc",
        "id": "abc",
        "name": "abc"
      }
    ],
    "chokePoints": [
      {
        "id": "abc",
        "name": "abc"
      }
    ],
    "hierarchy": "abc",
    "lat": -999,
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

404

エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

コード	説明
-----	----

500

内部サーバーエラーです

値の例 | モデル

```
{  
  "code": "ERROR_CODE",  
  "message": "error message",  
  "title": "error title",  
  "httpStatus": 0  
}
```

PUT

/entities/access/assets/{id}



この API は、ID を使用してアセットを更新するために使用されます。要求本文には、更新したい属性のみを指定してください。

パラメータ

試してみる

名前	説明
----	----

名前

説明

body * 必須

object
(本文)

アセット更新の要求本文：詳細は「UpdatedAsset」スキーマを参照してください。更新するアセットを取得するときは、要求本文に応答本文と同じ属性がすべて含まれていることを確認してください。欠落しているすべての属性は、システム内で null 値で上書きされ、現在のワークフローを妨げる可能性があります。「NewAsset」スキーマから指定された属性のみを変更します。

値の例 | モデル

```
{
  "id": 11,
  "serial": "serial-123",
  "name": "name-123",
  "description": "Asset Description",
  "status": "Active",
  "tags": [
    1
  ],
  "category": 100,
  "subcategories": [
    "subcategory 1"
  ],
  "department": 200,
  "customData": {
    "1": "test"
  },
  "categoryCustomData": {
    "1": "test"
  },
  "site": {
    "id": "abc",
    "level": "FLOOR",
```

パラメータコンテンツタイプ

application/json ▼

id * 必須

整数

(パス)

アセットの ID

id : アセットの ID

応答

応答コンテンツタイプ

application/json ▼

コード

説明

200

エンティティが正常に更新されました

コード 説明

400

要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

409

シリアル番号が一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

DELETE /entities/access/assets/{id}



この API は、ID を使用してアセットを削除するために使用されます。アセットを削除すると、割り当てられたタグのいずれかが「使用可能」としてマークされます。

パラメータ

試してみる

Name 説明

id * 必須

アセットの ID

整数

(パス)

id : アセットの ID

応答

応答コンテンツタイプ

application/json



コード

説明

200

エンティティが正常に削除されました

401

許可されていません

値の例 | モデル

```
Unauthorized
```

404

エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

POST

/entities/access/search/assets



この API は、フィルタを使用してアセットを取得するために使用されます。現在、アセットのシリアル番号またはタグの MAC アドレスを使用したフィルタがサポートされています。この API では、要求本文の一部として制限とページを指定する必要があります。この API が現在サポートしている上限は 100 です。ページはページ 0 から始まります。次の一連のアセットを取得するには、ページの値を増分するだけです。

パラメータ

[試してみる](#)

名前	説明
----	----

body * 必須

値の例 | モデル

object
(本文)

```
{
  "pagination": {
    "limit": 10,
    "page": 0
  },
  "fields": [
    {
      "key": "serial",
      "type": "TEXT",
      "operator": "contains",
      "values": [
        "serial123"
      ]
    }
  ]
}
```

パラメータコンテンツタイプ

application/json ▼

応答

応答コンテンツタイプ

application/json ▼

コード	説明
-----	----

201

フィルタを使用してすべてのアセットを正常に取得しました

値の例 | モデル

```
[
  {
    "count": 1,
    "data": [
      {
        "id": 1,
        "serial": "serial-123",
        "name": "name-123",
        "location": {
          "x": 100,
          "y": 100,
          "floorId": "abc",
          "zones": [
            {
              "color": "#abc",
              "id": "abc",
              "name": "abc"
            }
          ],
          "chokePoints": [
            {
              "id": "abc",
              "name": "abc"
            }
          ]
        }
      }
    ]
  }
]
```

400

要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

コード	説明
-----	----

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

カテゴリ



POST

/entities/access/categories



この API は、新しいカテゴリを作成するために使用されます。カテゴリを作成する前に、カテゴリに割り当てたい部門やカスタムフィールドの ID があることを確認してください。各エンティティには、システムに追加されるときに ID が割り当てられます。

パラメータ

試してみる

名前	説明
----	----

名前

説明

body * 必須

新規カテゴリ作成の要求本文：詳細は「NewCategory」スキーマを参照

object
(本文)

値の例 | モデル

```
{
  "name": "cat1",
  "description": "description for category",
  "subcategories": [
    "subcategory 1",
    "subcategory 2"
  ],
  "customData": {
    "1": "test"
  },
  "template": {
    "entityName": "CATEGORY_CUSTOM",
    "fields": [
      {
        "label": "Custom Field",
        "key": "customField",
        "entityName": "CATEGORY_CUSTOM",
        "description": "This is a custom field",
        "templateLabel": "Category Custom Field",
        "type": "SELECT",
        "options": [
          "option 1",
          "option 2"
        ]
      }
    ]
  }
}
```

パラメータコンテンツタイプ

application/json ▼

応答

応答コンテンツタイプ

application/json ▼

コード

説明

201

エンティティが正常に作成されました

値の例 | モデル

```
{
  "id": 0
}
```

コード 説明

400

要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

409

名前が一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/categories



この API は、カテゴリのリストを取得するために使用されます。この API では、クエリパラメータで制限とページを指定する必要があります。この API が現在サポートしている上限は 100 です。ページはページ 0 から始まります。次の一連のカテゴリを取得するには、ページの値を増分するだけです。

パラメータ

試してみる

名前 説明

limit * 必須

一度に取得できる数。100 が最大です。

整数

(クエリ)

```
limit : 一度に取得できる数。100 が最大です。
```

page * 必須

どのページから取得を開始するか。ページ 0 から始まります

整数

(クエリ)

```
page : どのページから取得を開始するか。ページ 0 から始まります
```

応答

応答コンテンツタイプ

application/json

コード

説明

200

すべてのカテゴリを正常に取得しました

値の例 | モデル

```
{}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/categories/{id}



この API は、ID を使用してカテゴリを取得するために使用されます。

パラメータ

試してみる

名前 説明

id * 必須

カテゴリの ID

整数

(パス)

id : カテゴリの ID

応答

応答コンテンツタイプ

application/json



コード

説明

200

単一のアセットを正常に取得しました

値の例 | モデル

```
{
  "id": 1,
  "name": "cat1",
  "description": "description for category",
  "subcategories": [
    "subcategory 1",
    "subcategory 2"
  ],
  "template": {
    "id": 1,
    "entityName": "CATEGORY_CUSTOM",
    "fields": [
      {
        "id": 100,
        "key": 100,
        "label": "custom field",
        "entityName": "CATEGORY_CUSTOM",
        "description": "description of field",
        "templateLabel": "category custom field",
        "type": "SELECT",
        "options": [
          "option 1",
          "option 2"
        ]
      }
    ]
  }
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

コード	説明
-----	----

404	エンティティが見つかりませんでした
-----	-------------------

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500	内部サーバーエラーです
-----	-------------

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

PUT /entities/access/categories/{id}



この API は、ID を使用してカテゴリを更新するために使用されます。要求本文には、更新したい属性のみを指定してください。

パラメータ

試してみる

名前	説明
----	----

名前

説明

body * 必須

object

(本文)

カテゴリ更新の要求本文：詳細は「UpdatedCategory」スキーマを参照してください。更新するアセットを取得するときは、要求本文に応答本文と同じ属性がすべて含まれていることを確認してください。欠落しているすべての属性は、システム内で null 値で上書きされ、現在のワークフローを妨げる可能性があります。「NewCategory」スキーマから指定された属性のみを変更します。

値の例 | モデル

```
{
  "id": 1,
  "name": "cat1",
  "description": "description for category",
  "subcategories": [
    "subcategory 1",
    "subcategory 2"
  ],
  "customData": {
    "1": "test"
  },
  "template": {
    "entityName": "CATEGORY_CUSTOM",
    "fields": [
      {
        "label": "Custom Field",
        "key": "customField",
        "entityName": "CATEGORY_CUSTOM",
        "description": "This is a custom field",
        "templateLabel": "Category Custom Field",
        "type": "SELECT",
        "options": [
          "option 1",
```

パラメータコンテンツタイプ

application/json ▼

id * 必須

カテゴリの ID

整数

(パス)

id : カテゴリの ID

応答

応答コンテンツタイプ

application/json ▼

コード

説明

200

エンティティが正常に更新されました

コード 説明

400

要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

409

名前が一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

DELETE

/entities/access/categories/{id}



ID を使用してカテゴリを削除します。カテゴリの削除は、カテゴリに関連付けられたアセットがない場合にのみ実行できます。

パラメータ

試してみる

名前 説明

id * 必須

カテゴリの ID

整数

(パス)

id : カテゴリの ID

応答

応答コンテンツタイプ

application/json



コード

説明

200

エンティティが正常に削除されました

400

アセットはまだこのエンティティに割り当てられています。削除できませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

404

エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

コード	説明
-----	----

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

部門



POST

/entities/access/departments



この API は、新しい部門を作成するために使用されます。部門を作成する前に、カテゴリに割り当てたいカスタムフィールドの ID があることを確認してください。各エンティティには、システムに追加されるときに ID が割り当てられます。

パラメータ

試してみる

名前

説明

body * 必須

新規部門作成の要求本文：詳細は「NewDepartment」スキーマを参照

object

値の例 | モデル

(本文)

```
{
  "name": "dept1",
  "customData": {
    "1": "test"
  },
  "description": "description for department"
}
```

パラメータコンテンツタイプ

application/json



応答

応答コンテンツタイプ

application/json



コード 説明

201 エンティティが正常に作成されました

値の例 | モデル

```
{
  "id": 0
}
```

400 要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401 許可されていません

値の例 | モデル

```
Unauthorized
```

409 名前が一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500 内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/departments



この API は、部門のリストを取得するために使用されます。この API では、クエリパラメータで制限とページを指定する必要があります。この API が現在サポートしている上限は 100 です。ページはページ 0 から始まります。次の一連の部門を取得するには、ページの値を増分するだけです。

パラメータ

[試してみる](#)

名前	説明
----	----

limit * 必須

一度に取得できる数。100 が最大です。

整数

(クエリ)

limit : 一度に取得できる数。100 が最大です。

page * 必須

どのページから取得を開始するか。ページ 0 から始まります

整数

(クエリ)

page : どのページから取得を開始するか。ページ 0 から始まります

応答

応答コンテンツタイプ

application/json

コード

説明

200

すべての部門を正常に取得しました

値の例 | モデル

```
[
  {
    "id": 1,
    "name": "dept1",
    "assetCount": 10,
    "categories": [
      {
        "id": 1,
        "name": "cat1",
        "description": "description for category",
        "subcategories": [
          "subcategory 1",
          "subcategory 2"
        ],
        "template": {
          "id": 1,
          "entityName": "CATEGORY_CUSTOM",
          "fields": [
            {
              "id": 100,
              "key": 100,
              "label": "custom field",
              "entityName": "CATEGORY_CUSTOM",
```

コード	説明
401	Unauthorized 値の例 モデル <pre>Unauthorized</pre>
500	内部サーバーエラーです 値の例 モデル <pre>{ "code": "ERROR_CODE", "message": "error message", "title": "error title", "httpStatus": 0 }</pre>

GET
/entities/access/departments/{id}
🔒

この API は、ID を使用して部門を取得するために使用されます。

パラメータ
試してみる

名前	説明
id * 必須 整数 (パス)	部門の ID <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px; display: flex; align-items: center;"> id : <input style="width: 100%;" type="text" value="部門の ID"/> </div>

応答
応答コンテンツタイプ application/json ▼

コード	説明
-----	----

コード

説明

200

単一の部門を正常に取得しました

値の例 | モデル

```
{
  "id": 1,
  "name": "dept1",
  "assetCount": 10,
  "categories": [
    {
      "id": 1,
      "name": "cat1",
      "description": "description for category",
      "subcategories": [
        "subcategory 1",
        "subcategory 2"
      ],
      "template": {
        "id": 1,
        "entityName": "CATEGORY_CUSTOM",
        "fields": [
          {
            "id": 100,
            "key": 100,
            "label": "custom field",
            "entityName": "CATEGORY_CUSTOM",
            "description": "description of field",
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

404

エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

コード	説明
-----	----

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

PUT

/entities/access/departments/{id}



この API は、ID を使用して部門を更新するために使用されます。要求本文には、更新したい属性のみを指定してください。

パラメータ

試してみる

名前	説明
----	----

body * 必須

object
(本文)

部門更新の要求本文：詳細は「UpdatedDepartment」スキーマを参照してください。更新するアセットを取得するときは、要求本文に応答本文と同じ属性がすべて含まれていることを確認してください。欠落しているすべての属性は、システム内で null 値で上書きされ、現在のワークフローを妨げる可能性があります。「NewDepartment」スキーマから指定された属性のみを変更します。

値の例 | モデル

```
{
  "id": 1,
  "name": "dept1",
  "customData": {
    "1": "test"
  },
  "description": "description for department"
}
```

パラメータコンテンツタイプ

application/json

id * 必須

整数
(パス)

部門の ID

id : 部門の ID

コード 説明

200 エンティティが正常に更新されました

400 要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401 許可されていません

値の例 | モデル

```
Unauthorized
```

409 名前が一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500 内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

DELETE

/entities/access/departments/{id}



この API は、ID を使用して部門を削除するために使用されます。部門の削除は、部門に関連付けられたアセットおよびカテゴリがない場合にのみ実行できます。

パラメータ

試してみる

名前 説明

id * 必須

部門の ID

整数

(パス)

id : 部門の ID

応答

応答コンテンツタイプ

application/json

コード 説明

200

エンティティが正常に削除されました

400

アセットはまだこのエンティティに割り当てられています。削除できませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

コード 説明

404

エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

フィールド



POST

/entities/access/fields



この API は、エンティティの新しいフィールドを作成するために使用されます。

名前	説明
----	----

body * 必須

新規フィールド作成の要求本文：詳細は「NewField」スキーマを参照

object

(本文)

値の例 | モデル

```
{
  "field": {
    "label": "Custom Field",
    "key": "customField",
    "entityName": "TAGS",
    "description": "This is a custom field",
    "templateLabel": "Tag Custom Field",
    "type": "SELECT",
    "options": [
      "option 1",
      "option 2"
    ],
    "multiple": false,
    "validation": "ANY",
    "editable": true,
    "general": true,
    "required": false
  }
}
```

パラメータコンテンツタイプ

application/json ▼

応答

応答コンテンツタイプ

application/json ▼

コード	説明
-----	----

201

エンティティが正常に作成されました

値の例 | モデル

```
{
  "id": 0
}
```


コード 説明

400 要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401 許可されていません

値の例 | モデル

```
Unauthorized
```

409 ラベルが一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500 内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/fields/templates/{entityName}



この API は、エンティティのすべてのカスタムフィールドを取得するために使用されます。現在のエンティティには、ASSETS、CATEGORIES、DEPARTMENTS、および TAGS が含まれます。

パラメータ

試してみる

名前 説明

entityName * 必須

すべてのフィールドを取得するエンティティ名 (ASSETS、CATEGORIES、DEPARTMENTS、TAGS)

文字列

(パス)

entityName : すべてのフィールドを取得するエンティティ名 (ASSETS)

応答

応答コンテンツタイプ

application/json



コード

説明

200

すべてのフィールドを正常に取得しました

値の例 | モデル

```
[
  {
    "id": 100,
    "entityName": "TAGS",
    "lastModified": "2017-03-17T04:43:34.156Z",
    "fields": [
      {
        "id": 100,
        "key": 100,
        "label": "custom field",
        "entityName": "TAGS",
        "description": "description of field",
        "templateLabel": "Tag custom field",
        "type": "SELECT",
        "options": [
          "option 1",
          "option 2"
        ],
        "collection": null,
        "validation": "ANY",
        "isDefault": true,
        "required": true,
        "templateUnique": true,
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

コード	説明
-----	----

500

エンティティが正常に作成されました

値の例 | モデル

```
{
  "id": 0
}
```

PUT

/entities/access/fields/{id}



この API は、ID を使用してカスタムフィールドを更新するために使用されます。要求本文には、更新したい属性のみを指定してください。

パラメータ

試してみる

名前	説明
----	----

body * 必須

object
(本文)

フィールド更新の要求本文：詳細については、「UpdatedField」スキーマを参照してください。更新するアセットを取得するときは、要求 本文に応答本文と同じ属性がすべて含まれていることを確認してください。欠落しているすべての属性は、システム内で null 値で上書きされ、現在のワークフローを妨げる可能性があります。「NewField」スキーマから指定された属性のみを変更します。

値の例 | モデル

```
{
  "id": 100,
  "label": "Custom Field",
  "description": "This is a custom field",
  "templateLabel": "Tag Custom Field",
  "type": "SELECT",
  "options": [
    "option 1",
    "option 2"
  ],
  "multiple": false,
  "required": false
}
```

パラメータコンテンツタイプ

application/json

id * 必須

整数

(パス)

フィールドの ID

id : フィールドの ID

応答

応答コンテンツタイプ

application/json



コード 説明

200 エンティティが正常に更新されました

400 要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401 許可されていません

値の例 | モデル

```
Unauthorized
```

409 ラベルが一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500 内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

DELETE

/entities/access/fields/{id}



この API は、ID を使用してフィールドを削除するために使用されます。

パラメータ

試してみる

名前 説明

id * 必須
フィールドの ID

整数

(パス)

id : フィールドの ID

応答

応答コンテンツタイプ

application/json

コード 説明

200
エンティティが正常に削除されました

401
許可されていません

値の例 | モデル

```
Unauthorized
```

404
エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500
内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

タグ

POST

/entities/access/tags



この API は、新しいタグを作成するために使用されます。

パラメータ

試してみる

名前

説明

body *必須

New Tag 作成のリクエストボディ : 詳細は「NewTag」スキーマを参照

object

値の例 | モデル

(本文)

```
{
  "serial": "11:22:33:44:55:66",
  "name": "Tag 1",
  "customData": {
    "1": "test"
  },
  "type": "RFID"
}
```

パラメータコンテンツタイプ

application/json ▼

応答

応答コンテンツタイプ

application/json ▼

コード

説明

201

エンティティが正常に作成されました

値の例 | モデル

```
{
  "id": 0
}
```

コード 説明

400

要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

409

MAC アドレスが一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/tags

この API は、タグのリストを取得するために使用されます。この API では、クエリパラメータで制限とページを指定する必要があります。この API が現在サポートしている上限は 100 です。ページはページ 0 から始まります。次の一連のタグを取得するには、ページの値を増分するだけです。

パラメータ

試してみる

名前 説明

limit *必須

一度に取得できる数。100 が最大です。

整数

(クエリ)

```
limit : 一度に取得できる数。100 が最大です。
```

page *必須

どのページから取得を開始するか。ページ 0 から始まります

整数

(クエリ)

```
page : どのページから取得を開始するか。ページ 0 から始まります
```

応答

応答コンテンツタイプ

application/json



コード 説明

200

すべてのタグを正常に取得しました

値の例 | モデル

```
[
  {
    "id": 1,
    "serial": "11:22:33:44:55:66",
    "name": "Tag 1",
    "data": [
      {}
    ],
    "assetId": 1,
    "type": "RFID",
    "location": {},
    "lastHeard": "2018-01-16T22:50:25.368Z"
  }
]
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```


コード	説明
-----	----

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/tags/unassigned

この API は、未割り当てのタグのリストを取得するために使用されます。この API では、クエリパラメータで制限とページを指定する必要があります。この API が現在サポートしている上限は 100 です。ページはページ 0 から始まります。次の一連の未割り当てのタグを取得するには、ページの値を増分するだけです。

パラメータ

試してみる

名前	説明
----	----

limit * 必須

一度に取得できる数。100 が最大です。

整数

(クエリ)

limit : 一度に取得できる数。100 が最大です。

page * 必須

どのページから取得を開始するか。ページ 0 から始まります

整数

(クエリ)

page : どのページから取得を開始するか。ページ 0 から始まります

応答

応答コンテンツタイプ

application/json



コード	説明
-----	----

コード 説明

200

すべての未割り当てのタグを正常に取得しました

値の例 | モデル

```
[
  {
    "id": 1,
    "serial": "11:22:33:44:55:66",
    "name": "Tag 1",
    "data": [
      {}
    ],
    "assetId": 1,
    "type": "RFID",
    "location": {},
    "lastHeard": "2018-01-16T22:50:25.368Z"
  }
]
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/entities/access/tags/{id}



この API は、ID を使用してタグを取得するために使用されます。

パラメータ

試してみる

名前

説明

名前 説明

id * 必須

タグの ID

整数

(パス)

id : タグの ID

応答

応答コンテンツタイプ

application/json



コード

説明

200

単一のタグを正常に取得しました

値の例 | モデル

```
{
  "id": 1,
  "serial": "11:22:33:44:55:66",
  "name": "Tag 1",
  "data": [
    {}
  ],
  "assetId": 1,
  "type": "RFID",
  "location": {},
  "lastHeard": "2018-01-16T22:50:25.368Z"
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

404

エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

コード	説明
-----	----

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

PUT

/entities/access/tags/{id}



この API は、ID を使用してタグを更新するために使用されます。要求本文には、更新したい属性のみを指定してください。

パラメータ

試してみる

名前	説明
----	----

body * 必須

object
(本文)

タグ更新の要求本文：詳細は「UpdatedTag」スキーマを参照してください。更新するアセットを取得するときは、要求 本文に応答本文と同じ属性がすべて含まれていることを確認してください。欠落しているすべての属性は、システム内で null 値で上書きされ、現在のワークフローを妨げる可能性があります。「NewTag」スキーマから指定された属性のみを変更します。

値の例 | モデル

```
{
  "id": 1,
  "serial": "11:22:33:44:55:66",
  "name": "Tag 1",
  "customData": {
    "1": "test"
  },
  "type": "RFID"
}
```

パラメータコンテンツタイプ

application/json

id * 必須

整数

(パス)

タグの ID

id : タグの ID

応答

応答コンテンツタイプ

application/json



コード 説明

200 エンティティが正常に更新されました

400 要求本文に無効な要求値が含まれています

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401 許可されていません

値の例 | モデル

```
Unauthorized
```

409 MAC アドレスが一意ではありません

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500 内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

DELETE

/entities/access/tags/{id}



この API は、ID を使用してタグを削除するために使用されます。タグは、アセットに割り当てられていない場合にのみ削除できます。

パラメータ

試してみる

名前	説明
----	----

id * 必須

タグの ID

整数

(パス)

id : タグの ID

応答

応答コンテンツタイプ

application/json



コード	説明
-----	----

200

エンティティが正常に削除されました

400

アセットはまだこのエンティティに割り当てられています。削除できませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

401

許可されていません

値の例 | モデル

```
Unauthorized
```

コード 説明

404 エンティティが見つかりませんでした

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

500 内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

履歴

GET

/history/access/location/{assetId}



この API は、ID を使用してアセットのロケーション履歴を取得するために使用されます。この API では、開始時刻と終了時刻を指定する必要があります。サポートされる開始時間と終了時間の最大間隔は 1 週間です。

パラメータ

試してみる

名前 説明

assetId * 必須

アセットの ID

整数

(パス)

assetId : アセットの ID

startTime * 必須

間隔の開始時間の UNIX タイムスタンプ。

整数

(クエリ)

startTime : 間隔の開始時刻の UNIX タイムスタンプ。

endtime * 必須

間隔の終了時間の UNIX タイムスタンプ。

整数

(クエリ)

endtime : 間隔の終了時刻の UNIX タイムスタンプ。

コード 説明

200 アセットのロケーション履歴を正常に取得しました

値の例 | モデル

```
{
  "assetId": 1,
  "tenantId": 1,
  "startTime": 1524009600,
  "endTime": 1524009600,
  "series": [
    {
      "data": [
        {
          "x": 100,
          "y": 100,
          "floorId": "abc",
          "zones": [
            {
              "color": "#abc",
              "id": "abc",
              "name": "abc"
            }
          ]
        }
      ],
      "chokePoints": [
        {
          "id": "abc",
          "name": "abc"
        }
      ]
    }
  ]
}
```

500 内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

GET

/history/access/telemetry/{assetId}



この API は、ID を使用してアセットのテレメトリ履歴を取得するために使用されます。この API では、開始時刻と終了時刻を指定する必要があります。サポートされる開始時間と終了時間の最大間隔は 1 週間です。

パラメータ

試してみる

名前

説明

assetId * 必須

アセットの ID

整数

(パス)

assetId : アセットの ID

startTime * 必須

間隔の開始時間の UNIX タイムスタンプ。

整数

(クエリ)

startTime : 間隔の開始時刻の UNIX タイムスタンプ。

endTime * 必須

間隔の終了時間の UNIX タイムスタンプ。

整数

(クエリ)

endTime : 間隔の終了時刻の UNIX タイムスタンプ。

tagMacAddress * 必須

アセットに割り当てられたタグの MAC アドレス。

文字列

(クエリ)

tagMacAddress : アセットに割り当てられたタグの Mac アドレス。

type * 必須

取得するテレメトリ履歴の種類 (TEMPERATURE または HUMIDITY)

文字列

(クエリ)

type : 取得するテレメトリ履歴の種類 (TEMPERATU

応答

応答コンテンツタイプ

application/json



コード

説明

コード 説明

200

アセットのテレメトリ履歴を正常に取得しました

値の例 | モデル

```
{
  "assetId": 1,
  "tenantId": 1,
  "startTime": 1524009600,
  "endTime": 1524009600,
  "type": "TEMPERATURE",
  "units": "Degree Celsius",
  "series": [
    {
      "data": [
        {
          "value": 10,
          "timestamp": "2018-04-19T07:01:04.411Z"
        }
      ]
    }
  ]
}
```

500

内部サーバーエラーです

値の例 | モデル

```
{
  "code": "ERROR_CODE",
  "message": "error message",
  "title": "error title",
  "httpStatus": 0
}
```

モデル

Asset > {

説明:

以下のプロパティはアセットを定義します。定義されていない応答に表示されるすべてのプロパティは、カスタムフィールドのキーおよび値と見なされます。

id

整数

例: 1

アセットの ID 番号

serial

文字列

例: serial-123

アセットの固有シリアル番号

name

文字列

例: name-123

アセットの名前

location

LocationData > {...}

description

文字列

例: description of asset

	<p>アセットの説明</p>
status	<p>文字列</p> <p>例: <i>Active</i></p> <p>「Active」または「Inactive」になります。アセットが現在イベントを受信するように設定されているかどうかを識別します</p>
tags	<p>▼ [</p> <p>アセットに関連付けられたタグオブジェクトの配列</p>
category	<p>Tag > {...}]</p> <p>Category > {...}</p>
subcategories	<p>▼ [</p> <p>例: <i>List ["subcategory 1"]</i></p> <p>アセットを分類するために選択されたサブカテゴリ</p>
department	<p>整数]</p> <p>department > {...}</p>
data	<p>▼ {</p> <p>説明:</p> <p>アセットのテレメトリおよびその他の任意のデータ</p> <p>}</p>
categoryCustomData	<p>▼ {</p> <p>説明:</p> <p>アセットに関連するカテゴリカスタム属性の値。キーはカスタムアセットフィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。</p> <p>}</p>
maps	<p>▼ [</p> <p>例: <i>List ["abc", "def", "ghi"]</i></p> <p>アセットの場所の階層</p>
lastHeard	<p>文字列]</p> <p>文字列</p> <p>例: <i>2018-01-16T22:50:25.368Z</i></p> <p>アセットがテレメトリの更新/イベントを最後に受信したときのタイムスタンプ</p>
lastModified	<p>文字列</p> <p>例: <i>2018-01-16T22:50:25.368Z</i></p> <p>アセットが最後に更新されたときのタイムスタンプ</p>
site	

<code>static</code>	AssetSite > {...} ブール値 <i>例: true</i> アセットが静的アセットかどうかを識別します
<code>staticLocation</code>	AssetStaticLocation > {...}
<code>telemetry</code>	▼ { 説明: アセットのテレメトリデータを定義する属性のセット }
<code>x</code>	number <i>例: 100</i> 静的アセットの x 座標
<code>y</code>	number <i>例: 100</i> 静的アセットの y 座標
<code>icon</code>	Icon > {...}
<code>image</code>	Image > {...}
	}

NewAsset ▼ {	
<code>serial*</code>	文字列 <i>例: serial-123</i> アセットの固有シリアル番号
<code>name</code>	文字列 <i>例: name-123</i> アセットの名前
<code>description</code>	文字列 <i>例: Asset Description</i> アセットの説明
<code>status</code>	文字列 <i>例: Active</i> アセットのステータス: 「Active」または「Inactive」から選択
<code>tags</code>	▼ [<i>例: List [1]</i> アセットに割り当てられたタグの ID。これらのタグを別のアセットに割り当てないでください。 整数]
<code>category*</code>	整数 <i>例: 100</i>

アセットのカテゴリ ID。カテゴリがアセットに割り当てられた部門に属していることを確認してください

subcategories

▼ [

例: `List ["subcategory"]`

アセットのサブカテゴリ。オプションは、アセットに対して選択されたカテゴリの下にあります。

文字列]

department*

整数

例: `200`

アセットの部門 ID

customData

▼ {

説明:

アセットのカスタム属性の値。キーはカスタムアセットフィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。

}

categoryCustomData

▼ {

説明:

アセットのカテゴリカスタム属性の値。キーはカスタムアセットフィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。

}

site

AssetSite > {...}

staticLocation

AssetStaticLocation > {...}

static

ブール値

例: `true`

アセットに静的な場所が指定されている場合は `true`

x

number

例: `100`

アセットの静的位置の x 座標

y

number

例: `100`

アセットの静的位置の y 座標

icon

```
    icon > {...}
    image
    Image > {...}
}
```

UpdatedAsset ▾ {

id* **整数**
例: 11
アセットの ID

serial **文字列**
例: serial-123
アセットの固有シリアル番号

name **文字列**
例: name-123
アセットの名前

description **文字列**
例: Asset Description
アセットの説明

status **文字列**
例: Active
アセットのステータス: 「Active」または「Inactive」から選択

tags ▾ [
例: List [1]
アセットに関連付けられたタグ ID の配列
整数]

category **整数**
例: 100
アセットのカテゴリ ID

subcategories ▾ [
例: List ["subcategory 1"]
アセットのサブカテゴリ
文字列]

department **整数**
例: 200
アセットの部門 ID

customData ▾ {
説明:

アセットのカスタム属性の値。キーはカスタムアセットフィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、任意のテキストを入力します。フィールドが

日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。定義されていないアセット応答オブジェクトのプロパティは、このオブジェクトに配置する必要があります。

```
categoryCustomData }
                    v {
                      説明 :
                    }
site
static
staticLocation
x
y
icon
image
}
```

AssetSite > {...}

ブール値

例: *true*

アセットが静的アセットかどうかを識別します

AssetStaticLocation > {...}

数値

例: *100*

静的アセットの x 座標

数値

例: *100*

静的アセットの y 座標

Icon > {...}

Image > {...}

AssetSite ▾ {

説明:

サイトがアセットに割り当てられている場合、そのサイトにアクセスできるユーザーのみがアクセスできます。サイトのプロパティは、マップベース階層のデータを使用して設定できます。

id*

文字列

例: *abc*

アセットに割り当てられたサイトレベルの ID 番号

level*

文字列

例: *FLOOR*

アセットに割り当てられたサイトレベル (CAMPUS、BUILDING、または FLOOR)

name*

文字列

例: *campus -> building -> floor*

アセットに割り当てられたサイト階層の名前

campus*

▽ [

例: *List ["abc"]*

アセットに割り当てられたキャンパスの ID

文字列]

building*

▽ [

例: *List ["def"]*

アセットに割り当てられた建物の ID

文字列]

floor*

▽ [

例: *List ["ghi"]*

アセットに割り当てられたフロアの ID

文字列]

}

AssetStaticLocation ▾ {



説明:

静的な場所がアセットに割り当てられている場合、アセットは新しい場所の更新を受け取りません。静的ロケーションのプロパティは、マップベース階層のデータを使用して設定できます。

id*

文字列

例: abc

アセットに割り当てられた静的ロケーションレベルの ID 番号

level*

文字列

例: FLOOR

アセットに割り当てられた静的ロケーションレベル (常に FLOOR である必要があります)

name*

文字列

例: campus -> building -> floor

アセットに割り当てられた静的ロケーション階層の名前

campus*

▽ [

例: List ["abc"]

アセットに割り当てられた静的キャンパスの ID

文字列]

building*

▽ [

例: List ["def"]

アセットに割り当てられた静的建物の ID

文字列]

floor*

▽ [

例: List ["ghi"]

アセットに割り当てられた静的フロアの ID

文字列]

}

FilteredAssets ▾ {



説明:

以下のプロパティは、フィルタを使用してアセットを取得するときの応答を定義します。フィルタに適合するアセットは、キー「データ」の下に配列として見つかります。

count

整数

例: 1

data

▽ [Asset > {...}]

filter

AssetFilter > {...}

}

AssetFilter ▾ {

説明:

以下のプロパティは、フィルタの構造を定義します。シリアル番号でアセットを検索するには、フィールドの 1 つとして「serial」をキーとして追加します。タグの MAC アドレスでアセットを検索するには、フィールドの 1 つとして「tags」をキーとして追加します。

pagination*

▾ {

limit*

整数

例: 10

一度に取得できる数。100 が最大です。

page*

整数

例: 0

どのページから取得を開始するか。ページ 0 から始まります

}

fields*

▾ [▾ {

key*

文字列

例: serial

アセットをフィルタリングする対象。

type*

文字列

例: TEXT

フィルタの種類。「TEXT」を使用

operator*

文字列

例: contains

「contains」または「equals」を使用

values*

▾ [文字列

例: serial123

アセットを検索する対象の値

]

}]

}

AssetLocationHistory ▾ {

説明:

以下のプロパティは、アセットのロケーション履歴を定義します

assetId	整数 例: 1
tenantId	整数 例: 1
startTime	整数 例: 1524009600
endTime	整数 例: 1524009600
series	▾ [LocationSeriesData > {...}]

}

LocationSeriesData ▾ {

data	▾ [LocationData > {...}]
------	--------------------------

}

LocationData ▾ {

説明:

アセットの場所を定義する属性のセット

x	整数 例: 100
y	整数 例: 100
floorId	文字列 例: abc
zones	▾ [Zone > {...}]
chokePoints	▾ [ChokePoint > {...}]
hierarchy	文字列 例: abc
lat	整数 例: -999
lng	整数 例: -999
timestamp	文字列

}

Zone ▾ {

color 文字列
例: #abc

id 文字列
例: abc

name 文字列
例: abc

}

ChokePoint ▾ {

id 文字列
例: abc

name 文字列
例: abc

}

AssetTelemetryHistory ▾ {

説明:

以下のプロパティは、アセットのテレメトリ履歴を定義します

assetId 整数
例: 1

tenantId 整数
例: 1

startTime 整数
例: 1524009600

endTime 整数
例: 1524009600

type 文字列
例: TEMPERATURE

units 文字列
例: Degree Celsius

series ▾ [TelemetrySeriesData > {...}]

}

TelemetrySeriesData ▾ {

data ▾ [TelemetryData > {...}]

}

TelemetryData ▾ {

value 整数
例: 10

timestamp 文字列
例: 2018-04-19T07:01:04.411Z

}

Error ▾ {

code* 文字列
例: ERROR_CODE

message* 文字列
例: error message

title 文字列
例: error title

httpStatus 整数

}

許可されていません 文字列

例: Unauthorized

Icon ▾ {

説明: アセット用に選択されたアイコンのプロパティ

name* 文字列

type* 文字列

}

Image ▾ {

説明: エンティティ用にアップロードされた画像の詳細。属性は、新しい画像を追加するときの応答本文と同じである必要があります。同じ画像をエンティティに割り当てたい場合は、同じ画像を再アップロードして、新しい画像の詳細をエンティティに割り当ててください。画像はアセットであり、カテゴリは現在サポートされています

name* 文字列

id* 文字列

type* 文字列

}

Category ▾ {

説明:

以下のプロパティは、カテゴリを定義します。定義されていない応答に表示されるすべてのプロパティは、カスタムフィールドと見なされます。

id

整数

例: 1

カテゴリの ID 番号

name

文字列

例: cat1

カテゴリの名前

description

文字列

例: description for category

カテゴリの説明

subcategories

▾ [

例: List ["subcategory 1", "subcategory 2"]

アセットを作成する際に選択できる、より特徴的なカテゴリのリスト

文字列]

template

GetCategoryTemplate > {...}

icon

Icon > {...}

assetCount

整数

例: 10

カテゴリに割り当てられたアセットの数

department

Department > {...}

lastModified

文字列

例: 2018-01-16T22:50:25.368Z

カテゴリが最後に更新された時間

image

Image > {...}

}



NewCategory ▾ {

name*

文字列

例: cat1

カテゴリの名前

description

文字列

例: description for category

カテゴリの説明

subcategories

▾ [

例: List ["subcategory 1", "subcategory 2"]

アセットを作成する際に選択できる、より特徴的なカテゴリのリスト

文字列]

customData

▾ {

説明:

カテゴリのカスタム属性の値

}

template

AddCategoryTemplate > {...}

icon

Icon > {...}

department*

整数

例: 1

カテゴリが割り当てられている部門 ID

image

Image > {...}

}

UpdatedCategory ▾ {

id* **整数**
例: 1
カテゴリの ID 番号

name* **文字列**
例: cat1
カテゴリの名前

description* **文字列**
例: description for category
カテゴリの説明

subcategories* ▾ [
例: List ["subcategory 1", "subcategory 2"]
アセットを作成する際に選択できる、より特徴的なカテゴリのリスト

customData* ▾ {
説明:
カテゴリのカスタム属性の値。キーはカスタムカテゴリフィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。定義されていないカテゴリ応答オブジェクトのプロパティは、このオブジェクトに配置する必要があります。

template* **AddCategoryTemplate > {...}**

icon* **Icon > {...}**

image* **Image > {...}**

department* **整数**
例: 1
カテゴリが割り当てられている部門 ID

}

AddCategoryTemplate ▾ {



説明:

カテゴリ用に作成されたカテゴリカスタムフィールド。ここに追加されたフィールドオブジェクトはすべて追加/更新されます。たとえば、テンプレートに元々カスタムフィールド「cf1」と「cf2」がありましたが、更新要求本文にはカスタムフィールド「cf1」と新しいカスタムフィールド「cf3」しか含まれていない場合、カスタムフィールド「cf2」はカテゴリから削除され、「cf3」が追加されます。カスタムフィールド「cf1」はカテゴリに残ります。

entityName

文字列

例: `CATEGORY_CUSTOM`

フィールドが属するエンティティ: 常に「`CATEGORY_CUSTOM`」である必要があります

fields

▽ [

カテゴリに追加するカスタムフィールドのリスト

CategoryCustomFieldProperties > {...}]

}

GetCategoryTemplate ▾ {



説明:

カテゴリ用に作成されたカテゴリカスタムフィールド

id

整数

例: `1`

カテゴリテンプレートの ID

entityName

文字列

例: `CATEGORY_CUSTOM`

フィールドが属するエンティティ: 常に「`CATEGORY_CUSTOM`」である必要があります

fields

▽ [

カテゴリに追加するカスタムフィールドのリスト

CategoryCustomField > {...}]

lastModified

文字列

例: `2018-01-16T22:50:25.368Z`

カテゴリテンプレートが最後に変更されたときのタイムスタンプ

}

CategoryCustomFieldProperties ▾ {



label*

文字列

例: Custom Field

ユーザーインターフェイスで表示されるフィールドのラベル

key*

文字列

例: customField

フィールドのキー: 通常はラベルの camel 形式バージョン。キーは最終的にフィールドの ID に変換されます

entityName*

文字列

例: CATEGORY_CUSTOM

フィールドが属するエンティティ (ASSETS、CATEGORIES、DEPARTMENTS、TAGS、CATEGORY_CUSTOM)。CATEGORY_CUSTOM は、カテゴリカスタムフィールド用です。カテゴリカスタムフィールドを追加するには、NewCategory の詳細に従ってください。

description*

文字列

例: This is a custom field

フィールドの説明。ユーザーインターフェイスに表示するときのヘルパーテキストとして使用されます

templateLabel*

文字列

例: Category Custom Field

CSV テンプレートでのフィールドの表示方法。通常、entityName (Tag、Category、Department、Asset) をフィールドラベルと連結することによって行われます。エンティティが CATEGORY_CUSTOM の場合、カテゴリ名をフィールドラベル (Cat1 Custom Field) と連結します。CATEGORY_CUSTOM エンティティは、カテゴリカスタムフィールド専用です。カテゴリカスタムフィールドを追加するには、NewCategory の詳細に従ってください。

type*

文字列

例: SELECT

フィールドのタイプ (TEXT、SELECT、TEL、DATE、PARAGRAPH)

options

▼ [

例: List ["option 1", "option 2"]

フィールドがサポートする文字列選択値の配列。フィールドのタイプが「SELECT」の場合にのみ必要です

文字列]

multiple

ブール値

例: false

フィールドに複数の値を選択できる場合。フィールドのタイプが「SELECT」の場合にのみ必要です

validation*

文字列

例: ANY

フィールド値に必要な検証の種類 (ANY、TEL)

editable*

ブール値

例: true

フィールドの値を編集できる場合。常に true

general*

ブール値

例: true

```
required*      フィールドが [General] タブに表示される必要がある場合。常に true  
              ブール値  
              例: false  
              フィールド値が必須な場合  
}
```

CategoryCustomField ▼ {

id* 整数
例: 100
フィールドの ID 番号

key* 文字列
例: 100
フィールドのキー。キーは最終的にフィールドの ID に変換されます

label* 文字列
例: custom field
ユーザーインターフェイスで表示されるフィールドのラベル

entityName* 文字列
例: CATEGORY_CUSTOM
フィールドが属するエンティティ

description* 文字列
例: description of field
フィールドの説明

templateLabel* 文字列
例: category custom field
CSV テンプレートでのフィールドの表示方法

type* 文字列
例: SELECT
フィールドのタイプ (TEXT、SELECT、TEL、DATE、PARAGRAPH)

options* ▼ [
例: List ["option 1", "option 2"]
フィールドがサポートする選択値
文字列]

collection* 文字列
例: null
フィールドがデータを入力するために使用するコレクション

validation* 文字列
例: ANY
フィールド値に必要な検証の種類 (ANY、TEL)

isDefault* ブール値

例: *true*

フィールドがデフォルトのフィールドの場合

required*

ブール値

例: *true*

フィールドに値が必要な場合

templateUnique*

ブール値

例: *true*

csv インポート中にフィールドの値が一意である必要がある場合

templateHidden*

ブール値

例: *false*

フィールドが csv テンプレートに表示される場合

hidden*

ブール値

例: *false*

ユーザーインターフェイスにフィールドが表示される場合

editable*

ブール値

例: *true*

フィールドの値を編集できる場合

general*

ブール値

例: *true*

フィールドが [General] タブに表示される必要がある場合

multiple*

ブール値

例: *false*

フィールドに複数の値を選択できる場合

defaultSelectAll*

ブール値

例: *false*

デフォルトでフィールドのすべての値を選択する必要がある場合

clickableLink*

ブール値

例: *false*

フィールドの値がクリック可能な場合

min*

整数

例: *1*

フィールドに入力できる最小値

max*

整数

例: *100*

フィールドに入力できる最大値

filterable*

ブール値

例: *true*

フィールドをエンティティのフィルタリングに使用できる場合

sortable*

ブール値

例: *true*

フィールドをエンティティの並べ替えに使用できる場合

```
}
```

Department ▾ {



説明:

以下のプロパティは部門を定義します。定義されていない応答に表示されるすべてのプロパティは、カスタムフィールドと見なされます。

id

整数

例: 1

部門の ID 番号

name

文字列

例: dept1

部署名

assetCount

整数

例: 10

部門に割り当てられたアセットの数

categories

▾ [

部門に割り当てられたカテゴリのリスト

description

Category > {...}]

文字列

例: description for department

部門の説明

lastModified

文字列

例: 2018-01-16T22:50:25.368Z

部門が最後に変更されたときのタイムスタンプ

```
}
```



NewDepartment ▾ {

name* **文字列**
例: dept1

部署名

customData ▾ {
 説明:

部門のカスタム属性の値。キーはカスタム部門フィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。

 }
description **文字列**
例: description for department

部門の説明

}



UpdatedDepartment ▾ {

id* **整数**
例: 1
部門の ID 番号

name* **文字列**
例: dept1
部署名

customData* ▾ {
説明:

部門のカスタム属性の値。キーはカスタム部門フィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。定義されていない部門応答オブジェクトのプロパティは、この中にオブジェクトに配置する必要があります。

description **文字列**
例: description for department
部門の説明

}

FieldTemplate ▾ {



id	整数 <i>例: 100</i> フィールドテンプレートの ID 番号。各カスタムフィールドにはテンプレートが割り当てられます。
entityName	文字列 <i>例: TAGS</i> フィールドが属するエンティティ (ASSETS、CATEGORIES、DEPARTMENTS、TAGS、CATEGORY_CUSTOM)。CATEGORY_CUSTOM は、カテゴリカスタムフィールド用です。カテゴリカスタムフィールドを追加するには、NewCategory の詳細に従ってください。
lastModified	文字列 <i>例: 2017-03-17T04:43:34.156Z</i> フィールドが最後に更新されたときのタイムスタンプ
fields	▾ [エンティティに属するすべてのフィールド Field > {...}
}	

Field ▾ {



id*	整数 <i>例: 100</i> フィールドの ID 番号
key*	文字列 <i>例: 100</i> フィールドのキー。キーは最終的にフィールドの ID に変換されます。
label*	文字列 <i>例: custom field</i> ユーザーインターフェイスで表示されるフィールドのラベル
entityName*	文字列 <i>例: TAGS</i> フィールドが属するエンティティ
description*	文字列 <i>例: description of field</i> フィールドの説明
templateLabel*	文字列 <i>例: Tag custom field</i> CSV テンプレートでのフィールドの表示方法
type*	文字列 <i>例: SELECT</i>

フィールドのタイプ (TEXT、SELECT、TEL、DATE、PARAGRAPH)

options* > [...]

collection* 文字列

例: null

フィールドがデータを入力するために使用するコレクション

validation* 文字列

例: ANY

フィールド値に必要な検証の種類 (ANY、TEL)

isDefault* ブール値

例: true

フィールドがデフォルトのフィールドの場合

required* ブール値

例: true

フィールドに値が必要な場合

templateUnique* ブール値

例: true

CSV インポート中にフィールドの値が一意である必要がある場合

templateHidden* ブール値

例: false

フィールドが CSV テンプレートに表示される場合

hidden* ブール値

例: false

ユーザーインターフェイスにフィールドが表示される場合

editable* ブール値

例: true

フィールドの値を編集できる場合

general* ブール値

例: true

フィールドが [General] タブに表示される必要がある場合

multiple* ブール値

例: false

フィールドに複数の値を選択できる場合

defaultSelectAll* ブール値

例: false

デフォルトでフィールドのすべての値を選択する必要がある場合

clickableLink* ブール値

例: false

フィールドの値がクリック可能な場合

min* 整数

例: 1

max*

フィールドに入力できる最小値

整数

例: 100

filterable*

フィールドに入力できる最大値

ブール値

例: true

sortable*

フィールドをエンティティのフィルタリングに使用できる場合

ブール値

例: true

フィールドをエンティティの並べ替えに使用できる場合

}

NewField ▾ {

field*

FieldProperties > {...}

}

FieldProperties ▾ {

説明:

フィールドのプロパティ

label*

文字列

例: Custom Field

ユーザーインターフェイスで表示されるフィールドのラベル

key*

文字列

例: customField

フィールドのキー: 通常はラベルの camel 形式バージョン。キーは最終的にフィールドの ID に変換されます。

entityName*

文字列

例: TAGS

フィールドが属するエンティティ (ASSETS、CATEGORIES、DEPARTMENTS、TAGS、CATEGORY_CUSTOM)。CATEGORY_CUSTOM は、カテゴリカスタムフィールド用です。カテゴリカスタムフィールドを追加するには、NewCategory の詳細に従ってください。

description*

文字列

例: This is a custom field

フィールドの説明。ユーザーインターフェイスに表示するときのヘルパーテキストとして使用されます

templateLabel*

文字列

例: Tag Custom Field

CSV テンプレートでのフィールドの表示方法。entityName (Tag、Category、Department、Asset) をフィールドラベルと連結することによってラベル付けされます。エンティティが CATEGORY_CUSTOM の場合、部門名とカテゴリ名をフィールドラベル (Dept1 Cat1 Custom Field) と連結します。CATEGORY_CUSTOM は、カテゴリカスタムフィールド用です。カテゴリカスタムフィールドを追加するには、NewCategory の詳細に従ってください。

type*

文字列

例: *SELECT*

フィールドのタイプ (TEXT、SELECT、TEL、DATE、PARAGRAPH)

options

▼ [

例: *List ["option 1", "option 2"]*

フィールドがサポートする文字列選択値の配列。フィールドのタイプが「SELECT」の場合にのみ必要です

multiple

文字列

ブール値

例: *false*

フィールドに複数の値を選択できる場合。フィールドのタイプが「SELECT」の場合にのみ必要です

validation*

文字列

例: *ANY*

フィールド値に必要な検証の種類 (ANY、TEL)

editable*

ブール値

例: *true*

フィールドの値を編集できる場合。常に *true*

general*

ブール値

例: *true*

フィールドが [General] タブに表示される必要がある場合。常に *true*

required*

ブール値

例: *false*

フィールド値が必須な場合

}

UpdatedField ▾ {

id*

整数

例: 100

フィールドの ID 番号

label

文字列

例: Custom Field

ユーザーインターフェイスで表示されるフィールドのラベル

description

文字列

例: This is a custom field

フィールドの説明。ユーザーインターフェイスに表示するときのヘルパーテキストとして使用されます

templateLabel

文字列

例: Tag Custom Field

CSV テンプレートでのフィールドの表示方法。entityName (Tag、Category、Department、Asset) をフィールドラベルと連結することによってラベル付けされます。エンティティが CATEGORY_CUSTOM の場合、部門名とカテゴリ名をフィールドラベル (Dept1 Cat1 Custom Field) と連結します。CATEGORY_CUSTOM は、カテゴリカスタムフィールド用です。カテゴリカスタムフィールドを追加するには、NewCategory の詳細に従ってください。

type

文字列

例: SELECT

フィールドのタイプ (TEXT、SELECT、TEL、DATE、PARAGRAPH)

options

▾ [

例: List ["option 1", "option 2"]

フィールドがサポートする文字列選択値の配列。フィールドのタイプが「SELECT」の場合にのみ必要です

文字列]

multiple

ブール値

例: false

フィールドに複数の値を選択できる場合。フィールドのタイプが「SELECT」の場合にのみ必要です

required

ブール値

例: false

フィールド値が必須な場合

}

Tag ▾ {

説明:

以下のプロパティは、タグを定義します。定義されていない応答に表示されるすべてのプロパティは、カスタムフィールドと見なされます。

id

整数

例: 1

タグの ID 番号

serial

文字列

例: 11:22:33:44:55:66

タグの一意の MAC アドレス

name

文字列

例: Tag 1

タグの名前

data

▽ [

例: List [OrderedMap {}]

タグのテレメトリデータ

> {...}]

assetId

整数

例: 1

タグに関連付けられているアセット ID

type

文字列

例: RFID

タグのタイプ (RFID、BLE、WIFI)

location

▽ {

説明:

タグの位置データ

}

lastHeard

文字列

例: 2018-01-16T22:50:25.368Z

最後にタグが確認されたときのタイムスタンプ

}



NewTag ▾ {

serial*

文字列

例: 11:22:33:44:55:66

タグの一意の MAC アドレス

name

文字列

例: Tag 1

タグの名前

customData

▾ {

説明:

タグのカスタム属性の値。キーはカスタムタグフィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。

}

type

文字列

例: RFID

タグのタイプ (RFID、BLE、WIFI)

}

UpdatedTag ▾ {

id* **整数**
例: 1
タグの ID 番号

serial **string**
例: 11:22:33:44:55:66
タグの一意の MAC アドレス

name **string**
例: Tag 1
タグの名前

customData ▾ {

説明:

タグのカスタム属性の値。キーはカスタムタグフィールドの ID ですが、値はカスタムフィールドのタイプに依存します。フィールドがテキストフィールドの場合は、テキストを入力します。フィールドが日付フィールドの場合は、タイムスタンプ文字列を入力します。フィールドが選択フィールドの場合は、いずれかの選択オプションのテキストを指定します。カスタムデータを更新する場合は、すべての新旧のキー指定し、キーと値のペアをここで更新してください。ここに含まれるデータは、すべてのカスタムデータを上書きします。

type **文字列**
例: RFID
タグのタイプ (RFID、BLE、WIFI)

}