

ポイント・ルート・スポット判断基準

GeoReferencerアプリケーションで使用するJSONデータの自動分類基準を定義します。

1. ルートデータの判断基準

ルートとして判定される条件（以下のすべてを満たす場合）：

必須条件

- `routeInfo` オブジェクトが存在する
- `routeInfo.startPoint` が存在する
- `routeInfo.endPoint` が存在する
- `points` 配列が存在する

検証条件

- `points` 配列内に `type: "waypoint"` を持つ要素が1つ以上存在する
- 各waypoint要素は `imageX` と `imageY` プロパティを持つ
- 各waypoint要素は `id` プロパティを持たない

データ構造例

```
{
  "routeInfo": {
    "startPoint": "C-01",
    "endPoint": "C-05",
    "waypointCount": 3
  },
  "points": [
    {
      "type": "waypoint",
      "imageX": 100,
      "imageY": 200
    },
    {
      "type": "waypoint",
      "imageX": 150,
      "imageY": 250
    }
  ]
}
```

2. スポットデータの判断基準

スポットとして判定される条件（以下のいずれかを満たす場合）：

パターン1: `spots`配列を持つ場合

- `spots` 配列が存在する
- 配列内の要素が以下を満たす：
 - `name` プロパティを持つ（空文字列でない）
 - `imageX` と `imageY` プロパティを持つ
 - `id` プロパティを持たない

パターン2: 単一スポット（直接形式）

- `name` プロパティを持つ（空文字列でない）
- `imageX` と `imageY` プロパティを持つ
- `id` プロパティを持たない

データ構造例

```
// パターン1: spots配列
{
  "spots": [
    {
      "name": "展望台",
      "imageX": 300,
      "imageY": 400
    }
  ]
}

// パターン2: 単一スポット
{
  "name": "展望台",
  "imageX": 300,
  "imageY": 400
}
```

3. ポイントデータの判断基準

ポイント（座標マーカー）として判定される条件：

必須条件

- `points` 配列が存在する
- 配列内に `type` が `waypoint` でない要素が1つ以上存在する

検証条件

- 各ポイント要素は以下を持つ：
 - `id` または `name` プロパティ
 - `imageX` と `imageY` プロパティ
 - `type` プロパティが存在しないか、`waypoint` 以外の値

データ構造例

```
{
  "points": [
    {
      "id": "C-01",
      "name": "基準点A",
      "imageX": 50,
      "imageY": 100
    },
    {
      "id": "C-02",
      "name": "基準点B",
      "imageX": 200,
      "imageY": 300,
      "type": "reference"
    }
  ]
}
```

4. 従来形式のフォールバック判定

上記の判定に該当しない場合の補完的判定はしない

5. 判定優先順位

1. **ルート判定** (routeInfo + waypoint構造)
2. **スポット判定** (spots配列 または 単一name+座標)
3. **ポイント判定** (points配列で非waypoint要素)

6. カウント方法

ポイント数

- `points` 配列から `type !== "waypoint"` の要素のみカウント
- waypointはルートの中間点なのでポイント数に含めない

ルート数

- 判定されたルートファイル1つにつき1本としてカウント

スポット数

- `spots` 配列が存在する場合は配列の要素数
- 単一スポット形式の場合は1個としてカウント

7. 注意事項

- 同一ファイル内でポイント・ルート・スポットが混在する場合、優先順位に従って1つの種別として判定
- `type: "waypoint"` を持つ要素は常にルートの構成要素として扱う

- 判定不可能なファイルは警告ログを出力し、処理をスキップする