API Documentation (API Gateway)

NET-SKYPLOT

February 22, 2025

1 API Documentation: `/dems` Endpoint

1.1 Overview

The `/dems` endpoint allows users to retrieve available Digital Elevation Models (DEMs) for a given set of receiver coordinates. This is essential for selecting the appropriate terrain data for GNSS planning, especially in areas with varying topography.

- **Endpoint URL**: `/dems`
- **HTTP Method**: `POST`
- **Content Type**: `application/json`

1.2 Request Format

The request body must be a JSON object containing the `receivers` field, which is an array of receiver objects. Each receiver must include geographic coordinates (latitude and longitude).

1.2.1 Request Payload Structure

```
{
  "receivers": [
    {
      "coordinates": {
        "latitude": number, // Required: Latitude in decimal degrees (-90 to 90)
        "longitude": number // Required: Longitude in decimal degrees (-180 to 180)
      }
    },
    // ... additional receivers (optional)
  ]
}
```

1.2.2 Validation Rules

- **Receivers**:
  - Must be a non-empty array.
  - Each receiver must have a `coordinates` object with `latitude` and `longitude`.
  - Latitude must be between -90 and 90.
  - Longitude must be between -180 and 180.
- **Distance Constraint (for multi-receiver requests)**:
  - Receivers must not be more than 20 km apart. This ensures the DEM data is relevant for the planning area.

1.3 Response Format

The response contains the available DEM sources, each with their name, description, best resolution, and a list of available DEMs. A recommended DEM may also be provided.

1.3.1 Success Response Structure

- **Status Code**: `200 OK`
- **Body**:

```
{
  "available_sources": {
    "source_key": {
      "name": "string",
      "description": "string",
      "best_resolution": number,
      "dems": [
        {
          "type": "string",
          "description": "string",
          "resolution": number
        },
        // ... additional DEMs
      ]
    },
    // ... additional sources
  },
  "recommended": "string" // or null
}
```

1.3.2 Error Response Structure

- **Status Code**: `400 Bad Request` or `503 Service Unavailable`
- **Body**:

```
{
  "error": "string" // Descriptive error message
}
```

1.4 Examples

1.4.1 Example 1: Single Receiver

- **Request**:

```
{
  "receivers": [
    {
      "coordinates": {
        "latitude": 37.7749,
        "longitude": -122.4194
      }
    }
  ]
}
```

- **Response** (200 OK):

```
{
  "available_sources": {
    "ot": {
      "name": "Open Topography",
      "description": "Global DEM service",
      "best_resolution": 30,
      "dems": [
        {"type": "SRTMGL3", "description": "SRTM GL3 90m", "resolution": 90},
        {"type": "SRTMGL1", "description": "SRTM GL1 30m", "resolution": 30},
        {"type": "SRTMGL1_E", "description": "SRTM GL1 Ellipsoidal 30m", "
            ↪ resolution": 30},
        {"type": "AW3D30", "description": "ALOS World 3D 30m", "resolution": 30},
        {"type": "AW3D30_E", "description": "ALOS World 3D Ellipsoidal 30m", "
            ↪ resolution": 30},
        {"type": "SRTM15Plus", "description": "Global Bathymetry SRTM15+ V2.1", "
            ↪ resolution": 500},
        {"type": "NASADEM", "description": "NASADEM Global DEM", "resolution":
            ↪ 30},
        {"type": "COP30", "description": "Copernicus Global DSM 30m", "resolution
            ↪ ": 30},
        {"type": "COP90", "description": "Copernicus Global DSM 90m", "resolution
            ↪ ": 90},
        {"type": "EU_DTM", "description": "EU DTM 30m", "resolution": 30},
        {"type": "GEDI_L3", "description": "GEDI L3 DTM 1000m", "resolution":
            ↪ 1000},
        {"type": "GEBCOIceTopo", "description": "GEBCO Ice Topography 500m", "
            ↪ resolution": 500},
        {"type": "GEBCOSubIceTopo", "description": "GEBCO Sub-Ice Topography 500m
            ↪ ", "resolution": 500}
      ]
    },
    "pgp": {
      "name": "Piemote Geoportale",
      "description": "Regional DEM service",
      "best_resolution": 10,
      "dems": [
        {"type": "PGP10", "description": "Piemonte 10m DTM", "resolution": 10}
      ]
    }
  },
  "recommended": null
}
```


1.4.2 Example 2: Multi-Receiver (Within Distance Limit)

- **Request**:


```
{
  "receivers": [
    {
      "coordinates": {
        "latitude": 37.7749,
        "longitude": -122.4194
      }
    },
    {
      "coordinates": {
        "latitude": 37.7859,
```

```
        "longitude": -122.4364
      }
    }
  ]
}
```

- **Response** (200 OK):

```
{
  "available_sources": {
    "ot": {
      "name": "Open Topography",
      "description": "Global DEM service",
      "best_resolution": 30,
      "dems": [
        {"type": "SRTMGL3", "description": "SRTM GL3 90m", "resolution": 90},
        {"type": "SRTMGL1", "description": "SRTM GL1 30m", "resolution": 30},
        {"type": "SRTMGL1_E", "description": "SRTM GL1 Ellipsoidal 30m", "
            ↪ resolution": 30},
        {"type": "AW3D30", "description": "ALOS World 3D 30m", "resolution": 30},
        {"type": "AW3D30_E", "description": "ALOS World 3D Ellipsoidal 30m", "
            ↪ resolution": 30},
        {"type": "SRTM15Plus", "description": "Global Bathymetry SRTM15+ V2.1", "
            ↪ resolution": 500},
        {"type": "NASADEM", "description": "NASADEM Global DEM", "resolution":
            ↪ 30},
        {"type": "COP30", "description": "Copernicus Global DSM 30m", "resolution
            ↪ ": 30},
        {"type": "COP90", "description": "Copernicus Global DSM 90m", "resolution
            ↪ ": 90},
        {"type": "EU_DTM", "description": "EU DTM 30m", "resolution": 30},
        {"type": "GEDI_L3", "description": "GEDI L3 DTM 1000m", "resolution":
            ↪ 1000},
        {"type": "GEBCOIceTopo", "description": "GEBCO Ice Topography 500m", "
            ↪ resolution": 500},
        {"type": "GEBCOSubIceTopo", "description": "GEBCO Sub-Ice Topography 500m
            ↪ ", "resolution": 500}
      ]
    },
    "pgp": {
      "name": "Piemote Geoportale",
      "description": "Regional DEM service",
      "best_resolution": 10,
      "dems": [
        {"type": "PGP10", "description": "Piemonte 10m DTM", "resolution": 10}
      ]
    }
  },
  "recommended": null
}
```

1.5 Edge Cases and Error Handling

The following scenarios will result in a `400 Bad Request` error:

- Missing `receivers` field.
- `receivers` is not an array or is empty.

- Any receiver is missing `coordinates`, `latitude`, or `longitude`.
- Invalid latitude or longitude values (out of bounds).
- For multi-receiver requests, if any two receivers are more than 20 km apart.

If the downstream DEM service is unavailable, a `503 Service Unavailable` error is returned:

```
{
  "error": "DEM service unavailable"
}
```

1.6 Notes for Front-End Developers

- **Coordinate Precision**: Use at least 4 decimal places for latitude and longitude to ensure accuracy.
- **Distance Calculation**: The API uses the Haversine formula to calculate distances between receivers.
- **DEM Selection**: The response provides available DEMs for the area covering all receivers. Use `type` and `source_key` for planning.

2 API Documentation: `/plan` Endpoint

2.1 Overview

The `/plan` endpoint allows users to create a GNSS planning request, providing details such as start time, duration, DEM selection, constellations, and receiver configurations. This endpoint supports both single-receiver and multi-receiver (differential GNSS) applications, returning visibility, DOP metrics, and skyplot data.

- **Endpoint URL**: `/plan`
- **HTTP Method**: `POST`
- **Content Type**: `application/json`

2.2 Request Format

The request body must be a JSON object containing required fields and optional fields based on the application type.

2.2.1 Request Payload Structure

```
{
  "start_datetime": "string", // Required: ISO 8601 format (e.g., "2025-03-01T08:00:00
      ↪ Z")
  "duration_hours": number, // Required: Positive number (e.g., 24)
  "dem": { // Required: DEM selection from /dems
    "type": "string", // e.g., "SRTMGL1"
    "source": "string" // e.g., "ot"
  },
  "constellations": ["string"], // Required: Non-empty list (e.g., ["GPS", "GALILEO"])
  "receivers": [ // Required: Array of receivers
    {
      "id": "string", // Required: Unique identifier (e.g., "receiver_1")
      "role": "string", // Required for differential_gnss: "base" or "rover"
      "coordinates": {
        "latitude": number, // Required: -90 to 90
        "longitude": number, // Required: -180 to 180
```

```
      "height": number // Required: Height in meters
    },
    "obstacles": [ // Optional: Obstacle polygons
      {
        "vertices": [
          {"latitude": number, "longitude": number},
          // ... at least 3 vertices
        ],
        "height": number // Required: Height in meters
      }
    ]
  }
 ],
 "application": "string" // Optional: "single" (default) or "differential_gnss"
}
```

2.2.2 Validation Rules

- **Required Fields**: `start_datetime`, `duration_hours`, `dem`, `constellations`, `receivers`.
- **start_datetime**: Must be in ISO 8601 format.
- **duration_hours**: Must be a positive number.
- **dem**: Must have `type` and `source`, matching options from `/dems`.
- **constellations**: Must be a non-empty list of valid constellation names (e.g., "GPS", "GALILEO", "BEIDOU", "GLONASS").
- **receivers**:
  - Must be a non-empty array.
  - Each receiver requires `id`, `coordinates` (with `latitude`, `longitude`, `height`), and optionally `role` and `obstacles`.
  - Latitude: -90 to 90, Longitude: -180 to 180, Height: any number.
  - Obstacles: Each must have at least 3 vertices and a positive `height`.
  - Distance: Receivers must be within 20 km of each other (for multi-receiver requests).
- **application**:
  - Optional; defaults to "single".
  - Must be "single" or "differential_gnss".
  - For "single": Exactly 1 receiver, no `role` specified.
  - For "differential_gnss": At least 2 receivers, exactly 1 with `role`: "base", at least 1 with `role`: "rover".

2.3 Response Format

The response provides detailed planning data, including visibility, DOP metrics, and skyplot information, sampled every 30 minutes.

2.3.1 Success Response Structure

- **Status Code**: `200 OK`
- **Body**:

```
{
  "status": "success",
  "request_id": "string",
  "planning_details": {
    "start_datetime": "string",
    "duration_hours": number,
    "interval_minutes": 30,
    "application": "string"
  },
  "receivers": [
    {
      "id": "string",
      "role": "string", // Only for differential_gnss
      "coordinates": {
        "latitude": number,
        "longitude": number,
        "height": number
      },
      "visibility": {
        "constellation_name": {
          "satellite_count": [
            {"time": "string", "count": number},
            // ... every 30 minutes
          ]
        },
        // ... additional constellations
      },
      "dop": {
        "time": ["string", "..."],
        "gdop": [number, "..."],
        "pdop": [number, "..."],
        "hdop": [number, "..."],
        "vdop": [number, "..."]
      },
      "common_visibility": { // Only for rovers in differential_gnss
        "constellation_name": {
          "satellite_count": [
            {"time": "string", "count": number},
            // ... every 30 minutes
          ]
        },
        // ... additional constellations
      },
      "common_dop": { // Only for rovers in differential_gnss
        "time": ["string", "..."],
        "gdop": [number, "..."],
        "pdop": [number, "..."],
        "hdop": [number, "..."],
        "vdop": [number, "..."]
      },
      "skyplot_data": {
        "satellites": [
          {
            "constellation": "string",
            "satellite_id": "string",
            "trajectory": [
              {
```

```
              "time": "string",
              "azimuth": number,
              "elevation": number,
              "visible": boolean
            },
            // ... every 30 minutes
          ]
        },
        // ... additional satellites
      ]
    }
  },
  // ... additional receivers
 ]
}
```

2.3.2 Error Response Structure

- **Status Code**: `400 Bad Request` or `500 Internal Server Error`
- **Body**:

```
{
  "error": "string" // Descriptive error message
}
```

2.4 Examples

2.4.1 Example 1: Single Receiver

- **Request**:

```
{
  "start_datetime": "2025-03-01T08:00:00Z",
  "duration_hours": 2,
  "dem": {"type": "SRTMGL1", "source": "ot"},
  "constellations": ["GPS", "GALILEO"],
  "receivers": [
    {
      "id": "receiver_1",
      "coordinates": {
        "latitude": 40.712776,
        "longitude": -74.005974,
        "height": 15.0
      },
      "obstacles": [
        {
          "vertices": [
            {"latitude": 40.713000, "longitude": -74.005500},
            {"latitude": 40.713200, "longitude": -74.005700},
            {"latitude": 40.713100, "longitude": -74.005900}
          ],
          "height": 12
        }
      ]
    }
  ],
  "application": "single"
}
```

- **Response** (200 OK):


```
{
  "status": "success",
  "request_id": "abc123",
  "planning_details": {
    "start_datetime": "2025-03-01T08:00:00Z",
    "duration_hours": 2,
    "interval_minutes": 30,
    "application": "single"
  },
  "receivers": [
    {
      "id": "receiver_1",
      "coordinates": {
        "latitude": 40.712776,
        "longitude": -74.005974,
        "height": 15.0
      },
      "visibility": {
        "GPS": {
          "satellite_count": [
            {"time": "2025-03-01T08:00:00Z", "count": 5},
            {"time": "2025-03-01T08:30:00Z", "count": 6},
            {"time": "2025-03-01T09:00:00Z", "count": 5},
            {"time": "2025-03-01T09:30:00Z", "count": 4}
          ]
        },
        "GALILEO": {
          "satellite_count": [
            {"time": "2025-03-01T08:00:00Z", "count": 3},
            {"time": "2025-03-01T08:30:00Z", "count": 4},
            {"time": "2025-03-01T09:00:00Z", "count": 4},
            {"time": "2025-03-01T09:30:00Z", "count": 3}
          ]
        }
      },
      "dop": {
        "time": ["2025-03-01T08:00:00Z", "2025-03-01T08:30:00Z", "2025-03-01T09
            ↪ :00:00Z", "2025-03-01T09:30:00Z"],
        "gdop": [3.5, 3.4, 3.6, 3.7],
        "pdop": [2.8, 2.7, 2.9, 3.0],
        "hdop": [1.5, 1.4, 1.6, 1.7],
        "vdop": [2.3, 2.2, 2.3, 2.4]
      },
      "skyplot_data": {
        "satellites": [
          {
            "constellation": "GPS",
            "satellite_id": "G01",
            "trajectory": [
              {"time": "2025-03-01T08:00:00Z", "azimuth": 45.0, "elevation": 30.0,
                  ↪  "visible": true},
              {"time": "2025-03-01T08:30:00Z", "azimuth": 47.0, "elevation": 32.0,
                  ↪  "visible": true},
              {"time": "2025-03-01T09:00:00Z", "azimuth": 49.0, "elevation": 33.0,
                  ↪  "visible": true},
              {"time": "2025-03-01T09:30:00Z", "azimuth": 51.0, "elevation": 34.0,
                  ↪  "visible": true}
```

```
          ]
        },
        {
          "constellation": "GALILEO",
          "satellite_id": "E02",
          "trajectory": [
            {"time": "2025-03-01T08:00:00Z", "azimuth": 120.0, "elevation":
                ↪ 10.0, "visible": false},
            {"time": "2025-03-01T08:30:00Z", "azimuth": 122.0, "elevation":
                ↪ 12.0, "visible": true},
            {"time": "2025-03-01T09:00:00Z", "azimuth": 124.0, "elevation":
                ↪ 14.0, "visible": true},
            {"time": "2025-03-01T09:30:00Z", "azimuth": 126.0, "elevation":
                ↪ 16.0, "visible": true}
          ]
        }
      ]
    }
  }
 ]
}
```


2.4.2 Example 2: Differential GNSS (Multi-Receiver)

- **Request**:


```
{
  "start_datetime": "2025-03-01T08:00:00Z",
  "duration_hours": 2,
  "dem": {"type": "SRTMGL1", "source": "ot"},
  "constellations": ["GPS", "GALILEO"],
  "receivers": [
    {
      "id": "receiver_1",
      "role": "base",
      "coordinates": {
        "latitude": 40.712776,
        "longitude": -74.005974,
        "height": 15.0
      },
      "obstacles": [
        {
          "vertices": [
            {"latitude": 40.713000, "longitude": -74.005500},
            {"latitude": 40.713200, "longitude": -74.005700},
            {"latitude": 40.713100, "longitude": -74.005900}
          ],
          "height": 12
        }
      ]
    },
    {
      "id": "receiver_2",
      "role": "rover",
      "coordinates": {
        "latitude": 40.722776,
        "longitude": -74.015974,
        "height": 18.0
```

```
      },
      "obstacles": []
    }
  ],
  "application": "differential_gnss"
}
```


- **Response** (200 OK):


```
{
  "status": "success",
  "request_id": "xyz789",
  "planning_details": {
    "start_datetime": "2025-03-01T08:00:00Z",
    "duration_hours": 2,
    "interval_minutes": 30,
    "application": "differential_gnss"
  },
  "receivers": [
    {
      "id": "receiver_1",
      "role": "base",
      "coordinates": {
        "latitude": 40.712776,
        "longitude": -74.005974,
        "height": 15.0
      },
      "visibility": {
        "GPS": {
          "satellite_count": [
            {"time": "2025-03-01T08:00:00Z", "count": 5},
            {"time": "2025-03-01T08:30:00Z", "count": 6},
            {"time": "2025-03-01T09:00:00Z", "count": 5},
            {"time": "2025-03-01T09:30:00Z", "count": 4}
          ]
        },
        "GALILEO": {
          "satellite_count": [
            {"time": "2025-03-01T08:00:00Z", "count": 3},
            {"time": "2025-03-01T08:30:00Z", "count": 4},
            {"time": "2025-03-01T09:00:00Z", "count": 4},
            {"time": "2025-03-01T09:30:00Z", "count": 3}
          ]
        }
      },
      "dop": {
        "time": ["2025-03-01T08:00:00Z", "2025-03-01T08:30:00Z", "2025-03-01T09
            ↪ :00:00Z", "2025-03-01T09:30:00Z"],
        "gdop": [3.5, 3.4, 3.6, 3.7],
        "pdop": [2.8, 2.7, 2.9, 3.0],
        "hdop": [1.5, 1.4, 1.6, 1.7],
        "vdop": [2.3, 2.2, 2.3, 2.4]
      },
      "skyplot_data": {
        "satellites": [
          {
            "constellation": "GPS",
            "satellite_id": "G01",
```

```
        "trajectory": [
          {"time": "2025-03-01T08:00:00Z", "azimuth": 45.0, "elevation": 30.0,
              ↪ "visible": true},
          {"time": "2025-03-01T08:30:00Z", "azimuth": 47.0, "elevation": 32.0,
              ↪ "visible": true},
          {"time": "2025-03-01T09:00:00Z", "azimuth": 49.0, "elevation": 33.0,
              ↪ "visible": true},
          {"time": "2025-03-01T09:30:00Z", "azimuth": 51.0, "elevation": 34.0,
              ↪ "visible": true}
        ]
      }
    ]
  }
},
{
  "id": "receiver_2",
  "role": "rover",
  "coordinates": {
    "latitude": 40.722776,
    "longitude": -74.015974,
    "height": 18.0
  },
  "visibility": {
    "GPS": {
      "satellite_count": [
        {"time": "2025-03-01T08:00:00Z", "count": 4},
        {"time": "2025-03-01T08:30:00Z", "count": 5},
        {"time": "2025-03-01T09:00:00Z", "count": 5},
        {"time": "2025-03-01T09:30:00Z", "count": 4}
      ]
    },
    "GALILEO": {
      "satellite_count": [
        {"time": "2025-03-01T08:00:00Z", "count": 2},
        {"time": "2025-03-01T08:30:00Z", "count": 3},
        {"time": "2025-03-01T09:00:00Z", "count": 3},
        {"time": "2025-03-01T09:30:00Z", "count": 2}
      ]
    }
  },
  "common_visibility": {
    "GPS": {
      "satellite_count": [
        {"time": "2025-03-01T08:00:00Z", "count": 3},
        {"time": "2025-03-01T08:30:00Z", "count": 4},
        {"time": "2025-03-01T09:00:00Z", "count": 4},
        {"time": "2025-03-01T09:30:00Z", "count": 3}
      ]
    },
    "GALILEO": {
      "satellite_count": [
        {"time": "2025-03-01T08:00:00Z", "count": 2},
        {"time": "2025-03-01T08:30:00Z", "count": 3},
        {"time": "2025-03-01T09:00:00Z", "count": 3},
        {"time": "2025-03-01T09:30:00Z", "count": 2}
      ]
    }
  },
```

```
      "dop": {
        "time": ["2025-03-01T08:00:00Z", "2025-03-01T08:30:00Z", "2025-03-01T09
            ↪ :00:00Z", "2025-03-01T09:30:00Z"],
        "gdop": [4.0, 3.9, 3.8, 3.9],
        "pdop": [3.2, 3.1, 3.0, 3.1],
        "hdop": [1.8, 1.7, 1.6, 1.7],
        "vdop": [2.6, 2.5, 2.4, 2.5]
      },
      "common_dop": {
        "time": ["2025-03-01T08:00:00Z", "2025-03-01T08:30:00Z", "2025-03-01T09
            ↪ :00:00Z", "2025-03-01T09:30:00Z"],
        "gdop": [3.8, 3.7, 3.6, 3.7],
        "pdop": [3.0, 2.9, 2.8, 2.9],
        "hdop": [1.6, 1.5, 1.4, 1.5],
        "vdop": [2.4, 2.3, 2.2, 2.3]
      },
      "skyplot_data": {
        "satellites": [
          {
            "constellation": "GPS",
            "satellite_id": "G01",
            "trajectory": [
              {"time": "2025-03-01T08:00:00Z", "azimuth": 46.0, "elevation": 29.0,
                  ↪  "visible": true},
              {"time": "2025-03-01T08:30:00Z", "azimuth": 48.0, "elevation": 31.0,
                  ↪  "visible": true},
              {"time": "2025-03-01T09:00:00Z", "azimuth": 50.0, "elevation": 32.0,
                  ↪  "visible": true},
              {"time": "2025-03-01T09:30:00Z", "azimuth": 52.0, "elevation": 33.0,
                  ↪  "visible": true}
            ]
          }
        ]
      }
    }
  ]
}
```


2.4.3 Example 3: Invalid Application (Multiple Receivers as Single)

- **Request**:


```
{
  "start_datetime": "2025-03-01T08:00:00Z",
  "duration_hours": 2,
  "dem": {"type": "SRTMGL1", "source": "ot"},
  "constellations": ["GPS", "GALILEO"],
  "receivers": [
    {
      "id": "receiver_1",
      "coordinates": {
        "latitude": 40.712776,
        "longitude": -74.005974,
        "height": 15.0
      }
    },
    {
      "id": "receiver_2",
```

```
      "coordinates": {
        "latitude": 40.722776,
        "longitude": -74.015974,
        "height": 18.0
      }
    }
  ],
  "application": "single"
}
```

- **Response** (400 Bad Request):

```
{
  "error": "Single receiver application requires exactly one receiver"
}
```

2.4.4 Example 4: Missing Base Role in Differential GNSS

- **Request**:

```
{
  "start_datetime": "2025-03-01T08:00:00Z",
  "duration_hours": 2,
  "dem": {"type": "SRTMGL1", "source": "ot"},
  "constellations": ["GPS", "GALILEO"],
  "receivers": [
    {
      "id": "receiver_1",
      "role": "rover",
      "coordinates": {
        "latitude": 40.712776,
        "longitude": -74.005974,
        "height": 15.0
      }
    },
    {
      "id": "receiver_2",
      "role": "rover",
      "coordinates": {
        "latitude": 40.722776,
        "longitude": -74.015974,
        "height": 18.0
      }
    }
  ],
  "application": "differential_gnss"
}
```

- **Response** (400 Bad Request):

```
{
  "error": "Exactly one receiver must have the role 'base' for differential GNSS"
}
```

2.5 Edge Cases and Error Handling

The following scenarios result in a `400 Bad Request` error:

- Missing required fields (`start_datetime`, `duration_hours`, etc.).
- Invalid `start_datetime` format.
- Non-positive `duration_hours`.
- Invalid `dem` structure or values.
- Empty or invalid `constellations`.
- Invalid receiver data (e.g., missing `id`, out-of-bounds coordinates).
- Receivers more than 20 km apart.
- For "single": More or fewer than 1 receiver.
- For "differential_gnss": Fewer than 2 receivers, no base, no rover, or multiple bases.

Unexpected errors result in a `500 Internal Server Error`:

```
{
  "error": "Internal server error: [details]"
}
```

2.6 Notes for Front-End Developers

- **Time Format**: All timestamps are in ISO 8601 UTC (e.g., "2025-03-01T08:00:00Z").
- **Sampling**: Data is sampled every 30 minutes (e.g., 4 points for 2 hours).
- **Skyplot**: Use `azimuth` (degrees, 0-360) and `elevation` (degrees, 0-90) for polar plots; `visible` indicates obstruction status.
- **Differential GNSS**: Use `common_visibility` and `common_dop` for rover accuracy analysis.