<https://gemini.google.com/u/1/app/2d30425fc8a5984a>

# **Database Views Specification (V2)**

## **1. Introduction**

This document outlines the specifications for database views created to support the Via di Francesco Pilgrimage Platform V2. Views are virtual tables whose contents are defined by a query. They are used to simplify complex queries, encapsulate logic, improve read performance for common data aggregations, and provide a stable interface for applications and APIs, even if underlying table structures change.

Each view specification will include:

* **View Name:** The fully qualified name of the view.
* **Purpose:** A brief description of why the view is needed and its primary use case.
* **DDL:** The CREATE VIEW statement.
* **Key Columns:** A description of important columns exposed by the view.
* **Usage Notes:** Any specific considerations for using the view, such as indexing on underlying tables or potential performance implications.

## **2. Module 2: Core Trail Hierarchy Views**

These views are designed to support common API queries related to trails, routes, and segments.

### **2.1. View:** public.routes\_summary\_view

* **Purpose:** To provide a summarized, easily queriable list of routes, including their essential identifying information, key statistics, and status. This view is primarily intended for use when listing routes associated with a specific trail or for general route browsing with minimal detail.
* **DDL:**

CREATE VIEW public.routes\_summary\_view AS

SELECT

r.id AS route\_id,

r.trail\_id,

r.name AS route\_name\_en, -- Assumes routes.name stores the English name

r.slug AS route\_slug,

r.short\_description AS route\_short\_description\_en, -- Assumes routes.short\_description is English

r.total\_distance\_km,

r.estimated\_duration\_days,

r.route\_difficulty,

r.operational\_status AS route\_operational\_status,

r.content\_visibility\_status AS route\_content\_visibility\_status,

r.is\_primary\_route\_for\_trail,

r.is\_featured\_route,

r.map\_overview\_media\_id, -- For a quick map thumbnail

r.deleted\_at AS route\_deleted\_at,

t.name AS trail\_name\_en, -- Parent trail's English name for context

t.slug AS trail\_slug

FROM

public.routes r

JOIN

public.trails t ON r.trail\_id = t.id;

COMMENT ON VIEW public.routes\_summary\_view IS 'Provides a summarized view of routes with key identifying information, statistics, and parent trail context. Intended for API listings and route browsing. Version: V2.';

COMMENT ON COLUMN public.routes\_summary\_view.route\_name\_en IS 'The English name of the route.';

COMMENT ON COLUMN public.routes\_summary\_view.route\_short\_description\_en IS 'The English short description of the route.';

COMMENT ON COLUMN public.routes\_summary\_view.total\_distance\_km IS 'Auto-calculated total distance of the route in kilometers.';

COMMENT ON COLUMN public.routes\_summary\_view.trail\_name\_en IS 'The English name of the parent trail.';

* **Key Columns:**
  + route\_id: Primary identifier for the route.
  + trail\_id: Identifier of the parent trail.
  + route\_name\_en: English name of the route.
  + route\_slug: URL-friendly slug for the route.
  + total\_distance\_km: Pre-calculated total distance.
  + route\_difficulty: Difficulty rating of the route.
  + route\_operational\_status: Current operational status of the route.
  + route\_content\_visibility\_status: Current editorial visibility of the route.
  + trail\_name\_en: English name of the parent trail, for context.
* **Usage Notes:**
  + This view assumes routes.name and routes.short\_description store the English (base language) versions. The API layer would be responsible for fetching translations if a different language is requested, using route\_id to query the public.translations table.
  + Performance relies on good indexing on public.routes (id, trail\_id, content\_visibility\_status, deleted\_at) and public.trails (id).
  + RLS policies applied to public.routes and public.trails will also apply to queries against this view.

### **2.2. View:** public.segments\_summary\_view

* **Purpose:** To offer a condensed list of segments for a given route, including their order, identifying information, and key statistics. This is useful for API endpoints that list segments belonging to a route without needing full geometric or all descriptive details.
* **DDL:**

CREATE VIEW public.segments\_summary\_view AS

SELECT

rs.route\_id,

s.id AS segment\_id,

rs.order\_in\_route,

s.name AS segment\_name\_en, -- Assumes segments.name stores the English name

s.slug AS segment\_slug,

s.short\_description AS segment\_short\_description\_en, -- Assumes segments.short\_description is English

s.distance\_km,

s.estimated\_walking\_time\_minutes,

s.elevation\_gain\_meters,

s.elevation\_loss\_meters,

s.segment\_difficulty,

s.operational\_status AS segment\_operational\_status,

s.content\_visibility\_status AS segment\_content\_visibility\_status,

s.deleted\_at AS segment\_deleted\_at,

s.start\_waypoint\_id,

s.end\_waypoint\_id

FROM

public.route\_segments rs

JOIN

public.segments s ON rs.segment\_id = s.id;

COMMENT ON VIEW public.segments\_summary\_view IS 'Provides a summarized list of segments for routes, including order, name, and key statistics. Intended for API endpoints listing segments of a route. Version: V2.';

COMMENT ON COLUMN public.segments\_summary\_view.segment\_name\_en IS 'The English name of the segment.';

COMMENT ON COLUMN public.segments\_summary\_view.segment\_short\_description\_en IS 'The English short description of the segment.';

COMMENT ON COLUMN public.segments\_summary\_view.distance\_km IS 'Auto-calculated distance of the segment in kilometers.';

* **Key Columns:**
  + route\_id: Identifier of the parent route.
  + segment\_id: Primary identifier for the segment.
  + order\_in\_route: Order of the segment within the route.
  + segment\_name\_en: English name of the segment.
  + distance\_km: Pre-calculated distance of the segment.
  + segment\_difficulty: Difficulty rating of the segment.
  + segment\_operational\_status: Current operational status of the segment.
  + segment\_content\_visibility\_status: Current editorial visibility of the segment.
* **Usage Notes:**
  + This view assumes segments.name and segments.short\_description store the English (base language) versions. The API layer would handle fetching translations using segment\_id.
  + Performance relies on indexing on public.route\_segments (route\_id, segment\_id, order\_in\_route) and public.segments (id, content\_visibility\_status, deleted\_at).
  + RLS policies applied to public.route\_segments and public.segments (and by extension, public.routes and public.trails for visibility checks) will apply.

### **Specification:** public.v\_trails\_detailed\_localized **(View)**

**1. Purpose & Primary Use-Cases**

* **Purpose:** To provide a comprehensive, denormalized, and localized representation of a single trail's core attributes, including its translated textual content and primary media details (logo, banner). Each row in the view represents a trail in a specific language.
* **Primary Use-Cases:**
  + Supplying data for the GET /trails/{trail\_id}?lang={language\_code} API endpoint, simplifying the backend logic required to fetch detailed trail information.
  + Providing a ready-to-use dataset for rendering a trail's main page on a website or mobile application in the user's selected language.
  + Facilitating internal reporting or data extraction that requires trail details along with their primary translations.

**2. View Definition (SQL DDL)**

SQL

CREATE OR REPLACE VIEW public.v\_trails\_detailed\_localized AS

SELECT

t.id AS trail\_id,

t.slug AS trail\_slug,

t.trail\_short\_code,

t.is\_featured,

t.operational\_status AS trail\_operational\_status,

t.content\_visibility\_status AS trail\_content\_visibility\_status,

t.deleted\_at AS trail\_deleted\_at,

t.estimated\_total\_distance\_km,

t.estimated\_total\_duration\_days,

t.overall\_difficulty,

t.official\_website\_url,

t.contact\_organization\_url,

t.created\_at AS trail\_created\_at,

t.updated\_at AS trail\_updated\_at,

cb.email AS created\_by\_email, -- Example: joining profiles for email

ub.email AS updated\_by\_email, -- Example: joining profiles for email

-- Language of this particular localized record for the trail's own fields

name\_trans.language\_code,

-- Translated Trail Fields (direct attributes of the trail)

name\_trans.translated\_text AS name,

alt\_names\_trans.translated\_text AS alternate\_names\_concat, -- Note: This would be one string if directly joined. Arrays of translations are harder in views without aggregation.

short\_desc\_trans.translated\_text AS short\_description,

full\_desc\_trans.translated\_text AS full\_description,

hist\_sig\_trans.translated\_text AS historical\_significance,

cult\_sig\_trans.translated\_text AS cultural\_significance,

pilgrim\_focus\_trans.translated\_text AS pilgrimage\_focus,

start\_point\_trans.translated\_text AS primary\_start\_point\_name,

end\_point\_trans.translated\_text AS primary\_end\_point\_name,

direction\_trans.translated\_text AS typical\_direction\_of\_travel,

waymarking\_desc\_trans.translated\_text AS waymarking\_description,

safety\_cons\_trans.translated\_text AS overall\_safety\_considerations,

key\_attr\_summary\_trans.translated\_text AS key\_attractions\_summary,

pilgrim\_cred\_info\_trans.translated\_text AS pilgrim\_credential\_info,

contact\_org\_name\_trans.translated\_text AS contact\_organization\_name,

data\_source\_credit\_trans.translated\_text AS primary\_data\_source\_credit,

data\_licence\_info\_trans.translated\_text AS data\_licence\_info,

general\_notes\_trans.translated\_text AS general\_notes\_for\_pilgrims,

meta\_desc\_seo\_trans.translated\_text AS meta\_description\_seo,

wp\_excerpt\_trans.translated\_text AS wordpress\_excerpt,

-- Best seasons to walk (TEXT[]) is tricky to translate directly per element in a simple view row.

-- This would typically be handled by the API layer fetching the array and then its translations.

-- Or the view could explode it, creating many more rows, which is usually not desired for a "detail" view.

t.best\_seasons\_to\_walk AS best\_seasons\_to\_walk\_en, -- Base English array

-- Logo Media Details

logo\_m.id AS logo\_media\_id,

logo\_m.storage\_object\_path\_original AS logo\_media\_original\_url, -- Example, adjust to actual media table columns

logo\_m.image\_variants\_json AS logo\_media\_variants,

logo\_m.media\_asset\_type AS logo\_media\_asset\_type,

logo\_m.media\_licence AS logo\_media\_licence,

logo\_alt\_trans.translated\_text AS logo\_media\_alt\_text,

-- Banner Media Details

banner\_m.id AS banner\_media\_id,

banner\_m.storage\_object\_path\_original AS banner\_media\_original\_url, -- Example

banner\_m.image\_variants\_json AS banner\_media\_variants,

banner\_m.media\_asset\_type AS banner\_media\_asset\_type,

banner\_m.media\_licence AS banner\_media\_licence,

banner\_alt\_trans.translated\_text AS banner\_media\_alt\_text

FROM

public.trails t

-- Profile joins for created/updated by (optional, can be fetched by ID if preferred)

LEFT JOIN public.profiles cb ON t.created\_by\_profile\_id = cb.id

LEFT JOIN public.profiles ub ON t.updated\_by\_profile\_id = ub.id

-- Join for Trail Name (drives the language for each view row for trail's own fields)

INNER JOIN public.translations name\_trans

ON name\_trans.table\_identifier = 'trails'

AND name\_trans.row\_foreign\_key = t.id::TEXT

AND name\_trans.column\_identifier = 'name'

AND name\_trans.translation\_status = 'published\_live' -- Assuming this status

-- Left Joins for other translatable fields of the trail entity itself

LEFT JOIN public.translations alt\_names\_trans ON alt\_names\_trans.table\_identifier = 'trails' AND alt\_names\_trans.row\_foreign\_key = t.id::TEXT AND alt\_names\_trans.column\_identifier = 'alternate\_names' AND alt\_names\_trans.language\_code = name\_trans.language\_code AND alt\_names\_trans.translation\_status = 'published\_live' -- Note: This is a simplified approach for text arrays. True per-element translation needs API logic.

LEFT JOIN public.translations short\_desc\_trans ON short\_desc\_trans.table\_identifier = 'trails' AND short\_desc\_trans.row\_foreign\_key = t.id::TEXT AND short\_desc\_trans.column\_identifier = 'short\_description' AND short\_desc\_trans.language\_code = name\_trans.language\_code AND short\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations full\_desc\_trans ON full\_desc\_trans.table\_identifier = 'trails' AND full\_desc\_trans.row\_foreign\_key = t.id::TEXT AND full\_desc\_trans.column\_identifier = 'full\_description' AND full\_desc\_trans.language\_code = name\_trans.language\_code AND full\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations hist\_sig\_trans ON hist\_sig\_trans.table\_identifier = 'trails' AND hist\_sig\_trans.row\_foreign\_key = t.id::TEXT AND hist\_sig\_trans.column\_identifier = 'historical\_significance' AND hist\_sig\_trans.language\_code = name\_trans.language\_code AND hist\_sig\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations cult\_sig\_trans ON cult\_sig\_trans.table\_identifier = 'trails' AND cult\_sig\_trans.row\_foreign\_key = t.id::TEXT AND cult\_sig\_trans.column\_identifier = 'cultural\_significance' AND cult\_sig\_trans.language\_code = name\_trans.language\_code AND cult\_sig\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations pilgrim\_focus\_trans ON pilgrim\_focus\_trans.table\_identifier = 'trails' AND pilgrim\_focus\_trans.row\_foreign\_key = t.id::TEXT AND pilgrim\_focus\_trans.column\_identifier = 'pilgrimage\_focus' AND pilgrim\_focus\_trans.language\_code = name\_trans.language\_code AND pilgrim\_focus\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations start\_point\_trans ON start\_point\_trans.table\_identifier = 'trails' AND start\_point\_trans.row\_foreign\_key = t.id::TEXT AND start\_point\_trans.column\_identifier = 'primary\_start\_point\_name' AND start\_point\_trans.language\_code = name\_trans.language\_code AND start\_point\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations end\_point\_trans ON end\_point\_trans.table\_identifier = 'trails' AND end\_point\_trans.row\_foreign\_key = t.id::TEXT AND end\_point\_trans.column\_identifier = 'primary\_end\_point\_name' AND end\_point\_trans.language\_code = name\_trans.language\_code AND end\_point\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations direction\_trans ON direction\_trans.table\_identifier = 'trails' AND direction\_trans.row\_foreign\_key = t.id::TEXT AND direction\_trans.column\_identifier = 'typical\_direction\_of\_travel' AND direction\_trans.language\_code = name\_trans.language\_code AND direction\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations waymarking\_desc\_trans ON waymarking\_desc\_trans.table\_identifier = 'trails' AND waymarking\_desc\_trans.row\_foreign\_key = t.id::TEXT AND waymarking\_desc\_trans.column\_identifier = 'waymarking\_description' AND waymarking\_desc\_trans.language\_code = name\_trans.language\_code AND waymarking\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations safety\_cons\_trans ON safety\_cons\_trans.table\_identifier = 'trails' AND safety\_cons\_trans.row\_foreign\_key = t.id::TEXT AND safety\_cons\_trans.column\_identifier = 'overall\_safety\_considerations' AND safety\_cons\_trans.language\_code = name\_trans.language\_code AND safety\_cons\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations key\_attr\_summary\_trans ON key\_attr\_summary\_trans.table\_identifier = 'trails' AND key\_attr\_summary\_trans.row\_foreign\_key = t.id::TEXT AND key\_attr\_summary\_trans.column\_identifier = 'key\_attractions\_summary' AND key\_attr\_summary\_trans.language\_code = name\_trans.language\_code AND key\_attr\_summary\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations pilgrim\_cred\_info\_trans ON pilgrim\_cred\_info\_trans.table\_identifier = 'trails' AND pilgrim\_cred\_info\_trans.row\_foreign\_key = t.id::TEXT AND pilgrim\_cred\_info\_trans.column\_identifier = 'pilgrim\_credential\_info' AND pilgrim\_cred\_info\_trans.language\_code = name\_trans.language\_code AND pilgrim\_cred\_info\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations contact\_org\_name\_trans ON contact\_org\_name\_trans.table\_identifier = 'trails' AND contact\_org\_name\_trans.row\_foreign\_key = t.id::TEXT AND contact\_org\_name\_trans.column\_identifier = 'contact\_organization\_name' AND contact\_org\_name\_trans.language\_code = name\_trans.language\_code AND contact\_org\_name\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations data\_source\_credit\_trans ON data\_source\_credit\_trans.table\_identifier = 'trails' AND data\_source\_credit\_trans.row\_foreign\_key = t.id::TEXT AND data\_source\_credit\_trans.column\_identifier = 'primary\_data\_source\_credit' AND data\_source\_credit\_trans.language\_code = name\_trans.language\_code AND data\_source\_credit\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations data\_licence\_info\_trans ON data\_licence\_info\_trans.table\_identifier = 'trails' AND data\_licence\_info\_trans.row\_foreign\_key = t.id::TEXT AND data\_licence\_info\_trans.column\_identifier = 'data\_licence\_info' AND data\_licence\_info\_trans.language\_code = name\_trans.language\_code AND data\_licence\_info\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations general\_notes\_trans ON general\_notes\_trans.table\_identifier = 'trails' AND general\_notes\_trans.row\_foreign\_key = t.id::TEXT AND general\_notes\_trans.column\_identifier = 'general\_notes\_for\_pilgrims' AND general\_notes\_trans.language\_code = name\_trans.language\_code AND general\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations meta\_desc\_seo\_trans ON meta\_desc\_seo\_trans.table\_identifier = 'trails' AND meta\_desc\_seo\_trans.row\_foreign\_key = t.id::TEXT AND meta\_desc\_seo\_trans.column\_identifier = 'meta\_description\_seo' AND meta\_desc\_seo\_trans.language\_code = name\_trans.language\_code AND meta\_desc\_seo\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations wp\_excerpt\_trans ON wp\_excerpt\_trans.table\_identifier = 'trails' AND wp\_excerpt\_trans.row\_foreign\_key = t.id::TEXT AND wp\_excerpt\_trans.column\_identifier = 'wordpress\_excerpt' AND wp\_excerpt\_trans.language\_code = name\_trans.language\_code AND wp\_excerpt\_trans.translation\_status = 'published\_live'

-- Join for Logo Media and its alt text translation

LEFT JOIN public.media logo\_m ON t.logo\_media\_id = logo\_m.id

LEFT JOIN public.translations logo\_alt\_trans

ON logo\_alt\_trans.table\_identifier = 'media' -- Assuming 'media' table\_identifier for media captions/alt\_text

AND logo\_alt\_trans.row\_foreign\_key = logo\_m.id::TEXT

AND logo\_alt\_trans.column\_identifier = 'alt\_text' -- Assuming 'alt\_text' as the column for media alt text

AND logo\_alt\_trans.language\_code = name\_trans.language\_code

AND logo\_alt\_trans.translation\_status = 'published\_live'

-- Join for Banner Media and its alt text translation

LEFT JOIN public.media banner\_m ON t.banner\_media\_id = banner\_m.id

LEFT JOIN public.translations banner\_alt\_trans

ON banner\_alt\_trans.table\_identifier = 'media'

AND banner\_alt\_trans.row\_foreign\_key = banner\_m.id::TEXT

AND banner\_alt\_trans.column\_identifier = 'alt\_text'

AND banner\_alt\_trans.language\_code = name\_trans.language\_code

AND banner\_alt\_trans.translation\_status = 'published\_live'

WHERE

-- Base filtering for view content (RLS will further refine)

t.deleted\_at IS NULL;

COMMENT ON VIEW public.v\_trails\_detailed\_localized IS 'Provides a comprehensive, denormalized, and localized representation of a single trail''s core attributes and primary media. Each row represents a trail in a specific language. Related collections like regions, routes, terrain/usage types are fetched separately. Version: V2.';

**3. Output Columns**

*(Abridged list - includes key non-translated, primary translated, and media fields. The DDL shows all output columns.)*

| **Column Name** | **Data Type** | **Description** |
| --- | --- | --- |
| trail\_id | BIGINT | The unique ID of the trail. |
| trail\_slug | TEXT | The URL-friendly slug of the trail. |
| is\_featured | BOOLEAN | Flag indicating if the trail is featured. |
| trail\_operational\_status | public.trail\_operational\_status\_enum | Operational status of the trail. |
| trail\_content\_visibility\_status | public.content\_visibility\_status\_enum | Visibility status of the base trail record. |
| language\_code | TEXT | The language code (e.g., 'en', 'it') for the translated textual fields in this row. |
| name | TEXT | The translated name of the trail in the specified language\_code. |
| short\_description | TEXT | The translated short description of the trail in the specified language\_code. |
| full\_description | TEXT | The translated full description of the trail in the specified language\_code. |
| ... (other translated fields) ... | TEXT | ... |
| best\_seasons\_to\_walk\_en | TEXT[] | Array of recommended seasons/months in English (base language). Individual element translation handled by API. |
| logo\_media\_id | BIGINT | ID of the logo media item. |
| logo\_media\_variants | JSONB | JSON object with URLs for different image variants of the logo. |
| logo\_media\_alt\_text | TEXT | Translated alt text for the logo in the specified language\_code. |
| banner\_media\_id | BIGINT | ID of the banner media item. |
| banner\_media\_variants | JSONB | JSON object with URLs for different image variants of the banner. |
| banner\_media\_alt\_text | TEXT | Translated alt text for the banner in the specified language\_code. |
| trail\_created\_at | TIMESTAMPTZ | Creation timestamp of the trail record. |
| trail\_updated\_at | TIMESTAMPTZ | Last update timestamp of the trail record. |

**4. Example Usage**

To get details for trail ID 1 in Italian:

SQL

SELECT

trail\_id,

trail\_slug,

name, -- This will be the Italian name

short\_description, -- Italian short description

logo\_media\_variants ->> 'thumbnail\_100x100' AS logo\_thumbnail\_url,

logo\_media\_alt\_text -- Italian alt text for logo

FROM

public.v\_trails\_detailed\_localized

WHERE

trail\_id = 1

AND language\_code = 'it'

AND trail\_content\_visibility\_status = 'published'; -- Assuming API applies this visibility

**5. Underlying Tables & Key Joins**

* Primary Table: public.trails (aliased as t)
* Key Joins:
  + public.translations (multiple LEFT JOINs, one INNER JOIN for name): For all translatable text fields of the trails entity itself.
  + public.media (aliased as logo\_m, banner\_m): For logo and banner image details.
  + public.translations (again, for alt\_text of media items).
  + public.profiles (aliased as cb, ub): Optional, for created/updated by user details.

**6. RLS (Row-Level Security) Considerations**

* The view should be defined with SECURITY INVOKER (default). RLS policies on the underlying public.trails, public.media, and public.translations tables will apply based on the querying user's permissions.
* The view's base WHERE clause (t.deleted\_at IS NULL) provides an initial filter. API queries should further filter by trail\_content\_visibility\_status = 'published' for public access.

**7. Performance & Optimization Notes**

* **Indexing on** public.translations**:** Absolutely critical. A composite index on (table\_identifier, column\_identifier, row\_foreign\_key, language\_code, translation\_status) is vital for performance due to multiple joins.
* **Indexing on** public.trails**:** Indexes on id (PK), slug, logo\_media\_id, banner\_media\_id, content\_visibility\_status, deleted\_at are important.
* **Indexing on** public.media**:** Index on id (PK).
* **Complexity:** This view involves many LEFT JOINs to the translations table. For languages with many missing translations, this is efficient. If all translations are usually present, performance should be monitored.
* **Materialized View Consideration:** If read performance for frequently accessed trails in specific languages becomes a bottleneck, this view could be a candidate for materialization, refreshed periodically.
* **Separate Collections:** Related collections (regions, terrain types, usage types, routes) are **not** aggregated into this view to keep its complexity manageable. The API layer should fetch these via separate, efficient queries filtered by trail\_id, potentially using other summary views (like public.routes\_summary\_view) or direct table queries.

**8. Assumptions & Dependencies**

* Assumes the public.translations table structure (with table\_identifier, column\_identifier, row\_foreign\_key, language\_code, translated\_text, translation\_status).
* Assumes translation\_status = 'published\_live' (or equivalent) is used to identify production-ready translations.
* Assumes media.image\_variants\_json and media.storage\_object\_path\_original (or similar) exist as per the public.media table spec.
* Assumes media translations for alt\_text use table\_identifier = 'media' and column\_identifier = 'alt\_text'.
* Translation of TEXT[] fields like best\_seasons\_to\_walk and alternate\_names (where each element is translatable): This view shows a simplified approach for alternate\_names\_concat (concatenating or taking the first if multiple exist for a language, which isn't ideal) and returns the base English array for best\_seasons\_to\_walk\_en. Robust per-element translation of arrays is better handled at the API/application layer after fetching the base array and its trail\_id.

**9. Next-Action Checklist**

* 🔴 **Finalize** translations **table linkage:** Confirm table\_identifier, column\_identifier, row\_foreign\_key usage, and translation\_status values for all translatable fields of trails and media.
* 🟠 **Implement and Test DDL:** Create the public.v\_trails\_detailed\_localized view in the database.
* 🟠 **Test Performance:** Thoroughly test query performance against this view, especially with various languages and data volumes.
* 🟢 **Optimize Joins:** Review join strategies if performance issues are identified. Consider if any LEFT JOIN to translations could be an INNER JOIN if a translation is mandatory for display.
* 🟢 **Document for API Consumers:** Clearly document the view's output, expected query patterns (especially filtering by language\_code and trail\_id), and the strategy for fetching related collections (like regions, routes, terrain/usage types).
* 🟢 **Consider API Layer Aggregation:** Re-confirm that aggregating related collections (terrain types, usage types, regions, routes) will be handled by the API layer making separate queries, rather than trying to embed them all as complex JSON arrays directly within this SQL view.

### **Specification:** public.v\_routes\_detailed\_localized **(View)**

**1. Purpose & Primary Use-Cases**

* **Purpose:** To provide a comprehensive, denormalized, and localized representation of a single route's attributes, including its translated textual content, key statistics, parent trail context, and primary associated media details (map, banner, GPX). Each row in the view represents a route in a specific language.
* **Primary Use-Cases:**
  + Supplying data for the GET /routes/{route\_id}?lang={language\_code} API endpoint, simplifying backend logic.
  + Rendering a route's main detail page on a website or mobile application in the user's selected language.
  + Internal reporting requiring detailed, localized route information.

**2. View Definition (SQL DDL)**

SQL

CREATE OR REPLACE VIEW public.v\_routes\_detailed\_localized AS

SELECT

r.id AS route\_id,

r.slug AS route\_slug,

r.trail\_id,

t.name AS trail\_name\_en, -- Parent trail's English name for context

t.slug AS trail\_slug,

r.route\_code,

r.is\_primary\_route\_for\_trail,

r.route\_category,

r.alternative\_to\_route\_id,

r.is\_featured\_route,

r.operational\_status AS route\_operational\_status,

r.content\_visibility\_status AS route\_content\_visibility\_status,

r.deleted\_at AS route\_deleted\_at,

r.total\_distance\_km, -- Auto-calculated

r.estimated\_duration\_days,

r.estimated\_total\_elevation\_gain\_meters, -- Auto-calculated

r.route\_difficulty,

r.start\_town\_id,

st.name AS start\_town\_name\_en, -- Example join to towns for English name

r.end\_town\_id,

et.name AS end\_town\_name\_en, -- Example join to towns for English name

r.created\_at AS route\_created\_at,

r.updated\_at AS route\_updated\_at,

rcb.email AS route\_created\_by\_email, -- Example

rub.email AS route\_updated\_by\_email, -- Example

-- Language of this particular localized record for the route's own fields

name\_trans.language\_code,

-- Translated Route Fields

name\_trans.translated\_text AS name,

alt\_name\_trans.translated\_text AS route\_alternate\_name,

short\_desc\_trans.translated\_text AS short\_description,

full\_desc\_trans.translated\_text AS full\_description,

theme\_focus\_trans.translated\_text AS route\_theme\_or\_focus,

curation\_source\_trans.translated\_text AS route\_curation\_source, -- V1 translatable

start\_desc\_trans.translated\_text AS start\_point\_description,

end\_desc\_trans.translated\_text AS end\_point\_description,

terrain\_summary\_trans.translated\_text AS terrain\_summary\_for\_route,

waymarking\_nav\_trans.translated\_text AS waymarking\_and\_navigation\_details,

public\_transit\_trans.translated\_text AS public\_transit\_at\_start\_end,

accessibility\_notes\_trans.translated\_text AS accessibility\_notes,

general\_notes\_trans.translated\_text AS general\_notes\_for\_route,

meta\_desc\_seo\_trans.translated\_text AS meta\_description\_seo,

wp\_excerpt\_trans.translated\_text AS wordpress\_excerpt,

r.best\_seasons\_for\_route AS best\_seasons\_for\_route\_en, -- Base English array

-- Map Overview Media Details

map\_m.id AS map\_overview\_media\_id,

map\_m.storage\_object\_path\_original AS map\_overview\_media\_original\_url,

map\_m.image\_variants\_json AS map\_overview\_media\_variants,

map\_alt\_trans.translated\_text AS map\_overview\_media\_alt\_text,

-- Banner Media Details

banner\_m.id AS banner\_media\_id,

banner\_m.storage\_object\_path\_original AS banner\_media\_original\_url,

banner\_m.image\_variants\_json AS banner\_media\_variants,

banner\_alt\_trans.translated\_text AS banner\_media\_alt\_text,

-- Overall GPX Media Details

gpx\_m.id AS overall\_gpx\_media\_id,

gpx\_m.storage\_object\_path\_original AS overall\_gpx\_file\_url, -- URL to the actual GPX file

gpx\_m.file\_name\_original AS overall\_gpx\_file\_name,

gpx\_m.file\_size\_bytes\_original AS overall\_gpx\_file\_size\_bytes

FROM

public.routes r

JOIN public.trails t ON r.trail\_id = t.id

LEFT JOIN public.towns st ON r.start\_town\_id = st.id -- For English start town name

LEFT JOIN public.towns et ON r.end\_town\_id = et.id -- For English end town name

LEFT JOIN public.profiles rcb ON r.created\_by\_profile\_id = rcb.id

LEFT JOIN public.profiles rub ON r.updated\_by\_profile\_id = rub.id

-- Join for Route Name (drives the language for each view row)

INNER JOIN public.translations name\_trans

ON name\_trans.table\_identifier = 'routes'

AND name\_trans.row\_foreign\_key = r.id::TEXT

AND name\_trans.column\_identifier = 'name'

AND name\_trans.translation\_status = 'published\_live'

-- Left Joins for other translatable fields of the route entity

LEFT JOIN public.translations alt\_name\_trans ON alt\_name\_trans.table\_identifier = 'routes' AND alt\_name\_trans.row\_foreign\_key = r.id::TEXT AND alt\_name\_trans.column\_identifier = 'route\_alternate\_name' AND alt\_name\_trans.language\_code = name\_trans.language\_code AND alt\_name\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations short\_desc\_trans ON short\_desc\_trans.table\_identifier = 'routes' AND short\_desc\_trans.row\_foreign\_key = r.id::TEXT AND short\_desc\_trans.column\_identifier = 'short\_description' AND short\_desc\_trans.language\_code = name\_trans.language\_code AND short\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations full\_desc\_trans ON full\_desc\_trans.table\_identifier = 'routes' AND full\_desc\_trans.row\_foreign\_key = r.id::TEXT AND full\_desc\_trans.column\_identifier = 'full\_description' AND full\_desc\_trans.language\_code = name\_trans.language\_code AND full\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations theme\_focus\_trans ON theme\_focus\_trans.table\_identifier = 'routes' AND theme\_focus\_trans.row\_foreign\_key = r.id::TEXT AND theme\_focus\_trans.column\_identifier = 'route\_theme\_or\_focus' AND theme\_focus\_trans.language\_code = name\_trans.language\_code AND theme\_focus\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations curation\_source\_trans ON curation\_source\_trans.table\_identifier = 'routes' AND curation\_source\_trans.row\_foreign\_key = r.id::TEXT AND curation\_source\_trans.column\_identifier = 'route\_curation\_source' AND curation\_source\_trans.language\_code = name\_trans.language\_code AND curation\_source\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations start\_desc\_trans ON start\_desc\_trans.table\_identifier = 'routes' AND start\_desc\_trans.row\_foreign\_key = r.id::TEXT AND start\_desc\_trans.column\_identifier = 'start\_point\_description' AND start\_desc\_trans.language\_code = name\_trans.language\_code AND start\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations end\_desc\_trans ON end\_desc\_trans.table\_identifier = 'routes' AND end\_desc\_trans.row\_foreign\_key = r.id::TEXT AND end\_desc\_trans.column\_identifier = 'end\_point\_description' AND end\_desc\_trans.language\_code = name\_trans.language\_code AND end\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations terrain\_summary\_trans ON terrain\_summary\_trans.table\_identifier = 'routes' AND terrain\_summary\_trans.row\_foreign\_key = r.id::TEXT AND terrain\_summary\_trans.column\_identifier = 'terrain\_summary\_for\_route' AND terrain\_summary\_trans.language\_code = name\_trans.language\_code AND terrain\_summary\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations waymarking\_nav\_trans ON waymarking\_nav\_trans.table\_identifier = 'routes' AND waymarking\_nav\_trans.row\_foreign\_key = r.id::TEXT AND waymarking\_nav\_trans.column\_identifier = 'waymarking\_and\_navigation\_details' AND waymarking\_nav\_trans.language\_code = name\_trans.language\_code AND waymarking\_nav\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations public\_transit\_trans ON public\_transit\_trans.table\_identifier = 'routes' AND public\_transit\_trans.row\_foreign\_key = r.id::TEXT AND public\_transit\_trans.column\_identifier = 'public\_transit\_at\_start\_end' AND public\_transit\_trans.language\_code = name\_trans.language\_code AND public\_transit\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations accessibility\_notes\_trans ON accessibility\_notes\_trans.table\_identifier = 'routes' AND accessibility\_notes\_trans.row\_foreign\_key = r.id::TEXT AND accessibility\_notes\_trans.column\_identifier = 'accessibility\_notes' AND accessibility\_notes\_trans.language\_code = name\_trans.language\_code AND accessibility\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations general\_notes\_trans ON general\_notes\_trans.table\_identifier = 'routes' AND general\_notes\_trans.row\_foreign\_key = r.id::TEXT AND general\_notes\_trans.column\_identifier = 'general\_notes\_for\_route' AND general\_notes\_trans.language\_code = name\_trans.language\_code AND general\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations meta\_desc\_seo\_trans ON meta\_desc\_seo\_trans.table\_identifier = 'routes' AND meta\_desc\_seo\_trans.row\_foreign\_key = r.id::TEXT AND meta\_desc\_seo\_trans.column\_identifier = 'meta\_description\_seo' AND meta\_desc\_seo\_trans.language\_code = name\_trans.language\_code AND meta\_desc\_seo\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations wp\_excerpt\_trans ON wp\_excerpt\_trans.table\_identifier = 'routes' AND wp\_excerpt\_trans.row\_foreign\_key = r.id::TEXT AND wp\_excerpt\_trans.column\_identifier = 'wordpress\_excerpt' AND wp\_excerpt\_trans.language\_code = name\_trans.language\_code AND wp\_excerpt\_trans.translation\_status = 'published\_live'

-- Map Overview Media

LEFT JOIN public.media map\_m ON r.map\_overview\_media\_id = map\_m.id

LEFT JOIN public.translations map\_alt\_trans

ON map\_alt\_trans.table\_identifier = 'media'

AND map\_alt\_trans.row\_foreign\_key = map\_m.id::TEXT

AND map\_alt\_trans.column\_identifier = 'alt\_text'

AND map\_alt\_trans.language\_code = name\_trans.language\_code

AND map\_alt\_trans.translation\_status = 'published\_live'

-- Banner Media

LEFT JOIN public.media banner\_m ON r.banner\_media\_id = banner\_m.id

LEFT JOIN public.translations banner\_alt\_trans

ON banner\_alt\_trans.table\_identifier = 'media'

AND banner\_alt\_trans.row\_foreign\_key = banner\_m.id::TEXT

AND banner\_alt\_trans.column\_identifier = 'alt\_text'

AND banner\_alt\_trans.language\_code = name\_trans.language\_code

AND banner\_alt\_trans.translation\_status = 'published\_live'

-- Overall GPX Media

LEFT JOIN public.media gpx\_m ON r.overall\_gpx\_media\_id = gpx\_m.id -- No alt text for GPX typically

WHERE

r.deleted\_at IS NULL;

COMMENT ON VIEW public.v\_routes\_detailed\_localized IS 'Provides comprehensive, denormalized, and localized details for a single route, including parent trail context and primary media. Each row represents a route in a specific language. Segments are fetched separately. Version: V2.';

3. Output Columns

(Abridged list - includes key non-translated, primary translated, parent trail, and media fields. The DDL shows all output columns.)

| **Column Name** | **Data Type** | **Description** |
| --- | --- | --- |
| route\_id | BIGINT | The unique ID of the route. |
| route\_slug | TEXT | The URL-friendly slug of the route. |
| trail\_id | BIGINT | ID of the parent trail. |
| trail\_name\_en | TEXT | English name of the parent trail. |
| trail\_slug | TEXT | Slug of the parent trail. |
| route\_operational\_status | public.trail\_operational\_status\_enum | Operational status of the route. |
| route\_content\_visibility\_status | public.content\_visibility\_status\_enum | Visibility status of the base route record. |
| total\_distance\_km | REAL | Auto-calculated total distance of the route. |
| estimated\_total\_elevation\_gain\_meters | INTEGER | Auto-calculated total elevation gain of the route. |
| language\_code | TEXT | The language code (e.g., 'en', 'it') for the translated textual fields in this row. |
| name | TEXT | The translated name of the route in the specified language\_code. |
| short\_description | TEXT | The translated short description of the route in the specified language\_code. |
| ... (other translated fields) ... | TEXT | ... |
| best\_seasons\_for\_route\_en | TEXT[] | Array of recommended seasons/months in English (base language). Individual element translation by API. |
| map\_overview\_media\_id | BIGINT | ID of the map overview media item. |
| map\_overview\_media\_variants | JSONB | Image variants for the map overview. |
| map\_overview\_media\_alt\_text | TEXT | Translated alt text for the map overview media in the specified language\_code. |
| banner\_media\_id | BIGINT | ID of the banner media item. |
| banner\_media\_variants | JSONB | Image variants for the banner. |
| banner\_media\_alt\_text | TEXT | Translated alt text for the banner media in the specified language\_code. |
| overall\_gpx\_media\_id | BIGINT | ID of the overall GPX media item. |
| overall\_gpx\_file\_url | TEXT | Direct URL to the original GPX file (from media.storage\_object\_path\_original). |
| overall\_gpx\_file\_name | TEXT | Original file name of the GPX track. |

**4. Example Usage**

To get details for route ID 20 in Italian:

SQL

SELECT

route\_id,

route\_slug,

name, -- Italian name

trail\_name\_en, -- English trail name for context

total\_distance\_km,

map\_overview\_media\_variants ->> 'display\_800w' AS map\_image\_url,

map\_overview\_media\_alt\_text -- Italian alt text for map

FROM

public.v\_routes\_detailed\_localized

WHERE

route\_id = 20

AND language\_code = 'it'

AND route\_content\_visibility\_status = 'published'; -- Assuming API applies this

**5. Underlying Tables & Key Joins**

* Primary Table: public.routes (aliased as r)
* Key Joins:
  + public.trails (as t): For parent trail context.
  + public.towns (as st, et): For start/end town names (English only in this view, translations via separate query if needed for towns).
  + public.translations (multiple LEFT JOINs, one INNER JOIN for name): For all translatable text fields of the routes entity and its primary media.
  + public.media (as map\_m, banner\_m, gpx\_m): For map, banner, and GPX file details.
  + public.profiles (as rcb, rub): Optional, for created/updated by user details.

**6. RLS (Row-Level Security) Considerations**

* View defined with SECURITY INVOKER (default). RLS policies on public.routes, public.trails, public.media, public.towns, and public.translations will apply.
* The view's base WHERE clause (r.deleted\_at IS NULL) provides initial filtering. API queries should further filter by route\_content\_visibility\_status = 'published' and ensure the parent trail is also visible.

**7. Performance & Optimization Notes**

* **Indexing on** public.translations**:** Crucial (composite index on (table\_identifier, column\_identifier, row\_foreign\_key, language\_code, translation\_status)).
* **Indexing on** public.routes**:** Indexes on id (PK), slug, trail\_id, media FKs, content\_visibility\_status, deleted\_at.
* **Indexing on** public.trails**:** Indexes on id (PK).
* **Indexing on** public.media**:** Index on id (PK).
* **Complexity:** Similar to v\_trails\_detailed\_localized, involves many joins. Performance monitoring is advised.
* **Materialized View:** Could be a candidate for materialization if read performance is critical for frequently accessed routes.
* **Segments:** The ordered list of segments for a route is **not** included in this view to avoid excessive complexity and large row sizes (if segments were exploded). Segments should be fetched via a separate API call using route\_id, potentially utilizing public.segments\_summary\_view or querying public.segments directly (and its translations/media).

**8. Assumptions & Dependencies**

* Assumes public.translations table structure and translation\_status = 'published\_live'.
* Assumes public.media table structure including image\_variants\_json, storage\_object\_path\_original, file\_name\_original, file\_size\_bytes\_original.
* Assumes alt\_text for media is translated via public.translations with table\_identifier = 'media'.
* Assumes towns.name provides the English name; full town localization would be via town-specific API/view.
* Translation of best\_seasons\_for\_route (TEXT[]) is handled by API layer.

**9. Next-Action Checklist**

* 🔴 **Finalize** translations **table linkage:** Confirm all table\_identifier and column\_identifier values for routes and media translatable fields.
* 🟠 **Implement and Test DDL:** Create the public.v\_routes\_detailed\_localized view.
* 🟠 **Test Performance:** Test with various languages, data volumes, and common filter conditions.
* 🟢 **Optimize Joins/Consider Materialization:** Based on performance testing.
* 🟢 **Document for API Consumers:** Detail output columns, query patterns, and the strategy for fetching the associated list of segments.
* 🟢 **API Layer for Segments:** Ensure the API layer efficiently fetches and integrates the ordered list of segments (summary or full) when a client requests route details with segments.

### **Specification:** public.v\_segments\_detailed\_localized **(View)**

**1. Purpose & Primary Use-Cases**

* **Purpose:** To provide a comprehensive, denormalized, and localized representation of a single segment's attributes. This includes its translated textual content, pre-calculated geometric statistics, linked waypoint information (summary), dominant terrain type details, and GPX file information. Each row in the view represents a segment in a specific language.
* **Primary Use-Cases:**
  + Supplying data for the GET /segments/{segment\_id}?lang={language\_code} API endpoint, simplifying backend logic for fetching detailed segment information.
  + Rendering a segment's detail page or section within a route display on a website or mobile application in the user's selected language.
  + Internal reporting or data analysis requiring detailed, localized segment data.

**2. View Definition (SQL DDL)**

SQL

CREATE OR REPLACE VIEW public.v\_segments\_detailed\_localized AS

SELECT

s.id AS segment\_id,

s.slug AS segment\_slug,

s.name AS segment\_name\_en, -- Base English name for reference

s.start\_waypoint\_id,

wp\_start.name AS start\_waypoint\_name\_en, -- English name of start waypoint

s.end\_waypoint\_id,

wp\_end.name AS end\_waypoint\_name\_en, -- English name of end waypoint

s.path\_geom, -- Consider if this should be systematically excluded by default in API and only included on demand

s.distance\_km,

s.estimated\_walking\_time\_minutes,

s.elevation\_gain\_meters,

s.elevation\_loss\_meters,

s.min\_elevation\_meters,

s.max\_elevation\_meters,

s.average\_gradient\_percentage,

s.elevation\_profile\_data,

s.segment\_difficulty,

s.dominant\_terrain\_type\_id,

dtm.code AS dominant\_terrain\_code,

dtm.icon\_identifier AS dominant\_terrain\_icon\_identifier,

s.sun\_exposure\_level,

s.recommended\_travel\_direction,

s.operational\_status AS segment\_operational\_status,

s.content\_visibility\_status AS segment\_content\_visibility\_status,

s.deleted\_at AS segment\_deleted\_at,

s.is\_detour\_for\_segment\_id,

s.gpx\_media\_id,

gpx\_m.storage\_object\_path\_original AS gpx\_file\_url,

gpx\_m.file\_name\_original AS gpx\_file\_name,

s.primary\_data\_source\_segment,

s.created\_at AS segment\_created\_at,

s.updated\_at AS segment\_updated\_at,

scb.email AS segment\_created\_by\_email, -- Example

sub.email AS segment\_updated\_by\_email, -- Example

-- Language of this particular localized record

name\_trans.language\_code,

-- Translated Segment Fields

name\_trans.translated\_text AS name,

short\_desc\_trans.translated\_text AS short\_description,

detailed\_desc\_trans.translated\_text AS detailed\_description\_notes,

waymarking\_notes\_trans.translated\_text AS waymarking\_on\_segment\_notes,

suitability\_notes\_trans.translated\_text AS segment\_suitability\_notes,

water\_notes\_trans.translated\_text AS water\_sources\_general\_notes,

resupply\_notes\_trans.translated\_text AS resupply\_options\_general\_notes,

cultural\_notes\_trans.translated\_text AS segment\_cultural\_historical\_notes,

emergency\_notes\_trans.translated\_text AS emergency\_access\_notes,

weather\_advice\_trans.translated\_text AS segment\_weather\_advice,

-- Translated Dominant Terrain Name

dtm\_name\_trans.translated\_text AS dominant\_terrain\_name

FROM

public.segments s

LEFT JOIN public.waypoints wp\_start ON s.start\_waypoint\_id = wp\_start.id

LEFT JOIN public.waypoints wp\_end ON s.end\_waypoint\_id = wp\_end.id

LEFT JOIN public.terrain\_types\_master dtm ON s.dominant\_terrain\_type\_id = dtm.id

LEFT JOIN public.media gpx\_m ON s.gpx\_media\_id = gpx\_m.id

LEFT JOIN public.profiles scb ON s.created\_by\_profile\_id = scb.id

LEFT JOIN public.profiles sub ON s.updated\_by\_profile\_id = sub.id

-- Join for Segment Name (drives the language for each view row)

INNER JOIN public.translations name\_trans

ON name\_trans.table\_identifier = 'segments'

AND name\_trans.row\_foreign\_key = s.id::TEXT

AND name\_trans.column\_identifier = 'name'

AND name\_trans.translation\_status = 'published\_live'

-- Left Joins for other translatable fields of the segment entity

LEFT JOIN public.translations short\_desc\_trans ON short\_desc\_trans.table\_identifier = 'segments' AND short\_desc\_trans.row\_foreign\_key = s.id::TEXT AND short\_desc\_trans.column\_identifier = 'short\_description' AND short\_desc\_trans.language\_code = name\_trans.language\_code AND short\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations detailed\_desc\_trans ON detailed\_desc\_trans.table\_identifier = 'segments' AND detailed\_desc\_trans.row\_foreign\_key = s.id::TEXT AND detailed\_desc\_trans.column\_identifier = 'detailed\_description\_notes' AND detailed\_desc\_trans.language\_code = name\_trans.language\_code AND detailed\_desc\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations waymarking\_notes\_trans ON waymarking\_notes\_trans.table\_identifier = 'segments' AND waymarking\_notes\_trans.row\_foreign\_key = s.id::TEXT AND waymarking\_notes\_trans.column\_identifier = 'waymarking\_on\_segment\_notes' AND waymarking\_notes\_trans.language\_code = name\_trans.language\_code AND waymarking\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations suitability\_notes\_trans ON suitability\_notes\_trans.table\_identifier = 'segments' AND suitability\_notes\_trans.row\_foreign\_key = s.id::TEXT AND suitability\_notes\_trans.column\_identifier = 'segment\_suitability\_notes' AND suitability\_notes\_trans.language\_code = name\_trans.language\_code AND suitability\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations water\_notes\_trans ON water\_notes\_trans.table\_identifier = 'segments' AND water\_notes\_trans.row\_foreign\_key = s.id::TEXT AND water\_notes\_trans.column\_identifier = 'water\_sources\_general\_notes' AND water\_notes\_trans.language\_code = name\_trans.language\_code AND water\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations resupply\_notes\_trans ON resupply\_notes\_trans.table\_identifier = 'segments' AND resupply\_notes\_trans.row\_foreign\_key = s.id::TEXT AND resupply\_notes\_trans.column\_identifier = 'resupply\_options\_general\_notes' AND resupply\_notes\_trans.language\_code = name\_trans.language\_code AND resupply\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations cultural\_notes\_trans ON cultural\_notes\_trans.table\_identifier = 'segments' AND cultural\_notes\_trans.row\_foreign\_key = s.id::TEXT AND cultural\_notes\_trans.column\_identifier = 'segment\_cultural\_historical\_notes' AND cultural\_notes\_trans.language\_code = name\_trans.language\_code AND cultural\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations emergency\_notes\_trans ON emergency\_notes\_trans.table\_identifier = 'segments' AND emergency\_notes\_trans.row\_foreign\_key = s.id::TEXT AND emergency\_notes\_trans.column\_identifier = 'emergency\_access\_notes' AND emergency\_notes\_trans.language\_code = name\_trans.language\_code AND emergency\_notes\_trans.translation\_status = 'published\_live'

LEFT JOIN public.translations weather\_advice\_trans ON weather\_advice\_trans.table\_identifier = 'segments' AND weather\_advice\_trans.row\_foreign\_key = s.id::TEXT AND weather\_advice\_trans.column\_identifier = 'segment\_weather\_advice' AND weather\_advice\_trans.language\_code = name\_trans.language\_code AND weather\_advice\_trans.translation\_status = 'published\_live'

-- Join for Translated Dominant Terrain Name

LEFT JOIN public.translations dtm\_name\_trans

ON dtm\_name\_trans.table\_identifier = 'terrain\_types\_master'

AND dtm\_name\_trans.row\_foreign\_key = dtm.id::TEXT

AND dtm\_name\_trans.column\_identifier = 'name'

AND dtm\_name\_trans.language\_code = name\_trans.language\_code

AND dtm\_name\_trans.translation\_status = 'published\_live'

WHERE

s.deleted\_at IS NULL;

COMMENT ON VIEW public.v\_segments\_detailed\_localized IS 'Provides comprehensive, denormalized, and localized details for a single segment, including geometric stats, waypoint info, dominant terrain, and GPX media. Each row is a segment in a specific language. Additional terrain and media gallery items are fetched separately. Version: V2.';

3. Output Columns

(Abridged list - includes key non-translated, primary translated, and related entity fields. The DDL shows all output columns.)

| **Column Name** | **Data Type** | **Description** |
| --- | --- | --- |
| segment\_id | BIGINT | The unique ID of the segment. |
| segment\_slug | TEXT | The URL-friendly slug of the segment. |
| segment\_name\_en | TEXT | Base English name of the segment. |
| start\_waypoint\_id | BIGINT | ID of the start waypoint. |
| start\_waypoint\_name\_en | TEXT | English name of the start waypoint (full waypoint details via separate API if needed). |
| end\_waypoint\_id | BIGINT | ID of the end waypoint. |
| end\_waypoint\_name\_en | TEXT | English name of the end waypoint. |
| path\_geom | GEOMETRY | The full 3D path geometry (LineStringZ). |
| distance\_km | REAL | Auto-calculated distance of the segment. |
| ... (other geometric stats) ... | ... | ... |
| elevation\_profile\_data | JSONB | Auto-generated data for elevation profile charts. |
| dominant\_terrain\_code | TEXT | Code of the dominant terrain type. |
| dominant\_terrain\_icon\_identifier | TEXT | Icon identifier for the dominant terrain. |
| segment\_operational\_status | public.trail\_operational\_status\_enum | Operational status of the segment. |
| segment\_content\_visibility\_status | public.content\_visibility\_status\_enum | Visibility status of the base segment record. |
| gpx\_media\_id | BIGINT | ID of the GPX media item. |
| gpx\_file\_url | TEXT | URL to the original GPX file. |
| language\_code | TEXT | The language code for the translated fields in this row. |
| name | TEXT | Translated name of the segment. |
| short\_description | TEXT | Translated short description of the segment. |
| ... (other translated fields) ... | TEXT | ... |
| dominant\_terrain\_name | TEXT | Translated name of the dominant terrain type. |

**4. Example Usage**

To get details for segment ID 501 in Italian:

SQL

SELECT

segment\_id,

segment\_slug,

name, -- Italian name

distance\_km,

dominant\_terrain\_name, -- Italian terrain name

gpx\_file\_url

FROM

public.v\_segments\_detailed\_localized

WHERE

segment\_id = 501

AND language\_code = 'it'

AND segment\_content\_visibility\_status = 'published'; -- Assuming API applies visibility

**5. Underlying Tables & Key Joins**

* Primary Table: public.segments (aliased as s)
* Key Joins:
  + public.waypoints (as wp\_start, wp\_end): For start/end waypoint names.
  + public.terrain\_types\_master (as dtm): For dominant terrain details.
  + public.media (as gpx\_m): For GPX file details.
  + public.translations (multiple LEFT JOINs, one INNER JOIN for name): For all translatable text fields of the segments entity and its dominant terrain.
  + public.profiles (as scb, sub): Optional, for user details.

**6. RLS (Row-Level Security) Considerations**

* View defined with SECURITY INVOKER (default). RLS policies on public.segments, public.waypoints, public.terrain\_types\_master, public.media, and public.translations will apply.
* The view's base WHERE clause (s.deleted\_at IS NULL) provides initial filtering. API queries should further filter by segment\_content\_visibility\_status = 'published' and ensure it's part of a visible route/trail.

**7. Performance & Optimization Notes**

* **Indexing:** Critical indexes on public.translations (composite), public.segments (id, slug, FKs, path\_geom GIST), public.waypoints (id), public.terrain\_types\_master (id), public.media (id).
* path\_geom **and** elevation\_profile\_data**:** These can be large. APIs should consider allowing clients to selectively include/exclude them (e.g., using SELECT column specification in Supabase PostgREST calls) to manage payload size. The geometric stats (distance\_km, etc.) are pre-calculated on segments which is a major performance benefit.
* **Materialized View:** Could be a candidate for materialization if reads are very frequent and data changes less often.
* **Separate Collections:** segment\_additional\_terrain\_types and segment\_media (gallery) are **not** aggregated into this view. They should be fetched by the API layer using separate, efficient queries filtered by segment\_id.

**8. Assumptions & Dependencies**

* Assumes public.translations structure and translation\_status = 'published\_live'.
* Assumes public.media contains GPX files with relevant metadata.
* Assumes terrain\_types\_master.name is the translatable field for terrain names.
* English names for waypoints are fetched; full waypoint localization would be via a waypoint-specific API/view.

**9. Next-Action Checklist**

* 🔴 **Finalize** translations **table linkage:** Confirm all table\_identifier and column\_identifier values for segments and terrain\_types\_master translatable fields.
* 🟠 **Implement and Test DDL:** Create the public.v\_segments\_detailed\_localized view.
* 🟠 **Test Performance:** Test with various languages, data volumes, and especially when path\_geom or elevation\_profile\_data are selected.
* 🟢 **Optimize Joins/Consider Materialization:** Based on performance testing.
* 🟢 **Document for API Consumers:** Detail output columns, query patterns (especially language and segment ID filtering), and the strategy for fetching related collections (additional terrain, media gallery).
* 🟢 **API Layer for Collections:** Ensure the API layer can efficiently fetch and integrate segment\_additional\_terrain\_types (with their localized names) and segment\_media (with localized captions/alt-text and image variants).