https://gemini.google.com/u/1/app/d423d51e9f40da78

View Specification: `public.v\_waypoint\_attraction\_details\_localized`

\* \* \* \* \*

1\. View Name

`public.v\_waypoint\_attraction\_details\_localized`

\* \* \* \* \*

2\. Purpose & Primary Use-Cases

- Purpose: To provide a comprehensive, denormalized, and localized representation of an attraction's details, including its type, amenities, descriptive notes, associated media, and religious service schedules.

- Primary Use-Cases:

- Power the API endpoint `/waypoints/{waypoint\_id}/attraction\_details` by simplifying backend query logic.

- Provide all necessary data for displaying a complete attraction detail page in a web or mobile application, with support for internationalization.

- Facilitate easier data consumption by frontend applications by pre-joining and structuring related information.

\* \* \* \* \*

3\. View Schema (Columns)

\*(This view will select from `attractions\_details` and join various other tables. The `all\_translations` column will contain translations for fields directly on `attractions\_details`. Referenced master data like attraction type, amenities, service types, etc., will have their English labels directly included and their full translation sets also available within nested JSON structures.)\*

| Column Name | Data Type | Description |

| `waypoint\_id` | `BIGINT` | Inherited from `attractions\_details`. Links to `public.waypoints(id)`. PK of the attraction detail. |

| `attraction\_type\_id` | `INTEGER` | Inherited from `attractions\_details`. FK to `public.attraction\_types\_master(id)`. |

| `attraction\_type\_code` | `TEXT` | From `attraction\_types\_master.code`. |

| `attraction\_type\_label` | `TEXT` | Primary reference language (English) label from `attraction\_types\_master.label`. |

| `attraction\_type\_icon\_identifier` | `TEXT` | From `attraction\_types\_master.icon\_identifier`. |

| `attraction\_type\_all\_translations` | `JSONB` | All translations for the `label` and `description` of the referenced `attraction\_types\_master` record, keyed by language code. |

| `detailed\_description` | `TEXT` | Primary reference language (English) text from `attractions\_details.detailed\_description`. |

| `historical\_significance\_notes` | `TEXT` | Primary reference language (English) text from `attractions\_details.historical\_significance\_notes`. |

| `cultural\_significance\_notes` | `TEXT` | Primary reference language (English) text from `attractions\_details.cultural\_significance\_notes`. |

| `spiritual\_significance\_notes` | `TEXT` | Primary reference language (English) text from `attractions\_details.spiritual\_significance\_notes`. |

| `associated\_historical\_figures\_text` | `TEXT[]` | Inherited from `attractions\_details`. Elements are primary reference language (English). (Individual elements translatable via `public.translations` if keyed appropriately). |

| `key\_historical\_events\_notes` | `TEXT` | Primary reference language (English) text from `attractions\_details.key\_historical\_events\_notes`. |

| `opening\_hours\_structured` | `JSONB` | Inherited from `attractions\_details.opening\_hours\_structured`. |

| `opening\_hours\_text\_notes` | `TEXT` | Primary reference language (English) text from `attractions\_details.opening\_hours\_text\_notes`. |

| `opening\_hours\_last\_verified\_at` | `TIMESTAMPTZ` | Inherited from `attractions\_details.opening\_hours\_last\_verified\_at`. |

| `entry\_fee\_details` | `TEXT` | Primary reference language (English) text from `attractions\_details.entry\_fee\_details`. |

| `guided\_tours\_info` | `TEXT` | Primary reference language (English) text from `attractions\_details.guided\_tours\_info`. |

| `audio\_guides\_info` | `TEXT` | Primary reference language (English) text from `attractions\_details.audio\_guides\_info`. |

| `photography\_allowed\_notes` | `TEXT` | Primary reference language (English) text from `attractions\_details.photography\_allowed\_notes`. |

| `accessibility\_details\_specific` | `TEXT` | Primary reference language (English) text from `attractions\_details.accessibility\_details\_specific`. |

| `all\_translations` | `JSONB` | All translations for translatable fields directly on `attractions\_details` (e.g., `detailed\_description`, `historical\_significance\_notes`, etc.), keyed by language code and then field name. |

| `visitor\_amenities` | `JSONB` | Array of JSON objects, each representing a visitor amenity. Includes `id`, `code`, `label` (English), `icon\_identifier`, and `all\_translations` for the amenity's label/description. |

| `media\_gallery` | `JSONB` | Array of JSON objects, each representing a media item. Includes `media\_id`, `media\_role\_code`, `alt\_text` (English), `caption` (English), `image\_variants\_json`, and `all\_translations` for alt/caption. |

| `religious\_services` | `JSONB` | Array of JSON objects, each representing a religious service schedule. Includes all fields from `religious\_service\_schedules` with nested translated types and languages. |

| `data\_last\_verified\_at` | `TIMESTAMPTZ` | Inherited from `attractions\_details.data\_last\_verified\_at`. |

| `created\_at` | `TIMESTAMPTZ` | Inherited from `attractions\_details.created\_at`. |

| `created\_by\_full\_name` | `TEXT` | Full name of the profile that created the attraction detail (joined from `public.profiles`). |

| `updated\_at` | `TIMESTAMPTZ` | Inherited from `attractions\_details.updated\_at`. |

| `updated\_by\_full\_name` | `TEXT` | Full name of the profile that last updated the attraction detail (joined from `public.profiles`). |

| `parent\_waypoint\_name` | `TEXT` | The `name` of the parent waypoint from `public.waypoints` (primary language). |

| `parent\_waypoint\_slug` | `TEXT` | The `slug` of the parent waypoint from `public.waypoints`. |

\* \* \* \* \*

4\. Underlying SQL Definition

SQL

```

CREATE OR REPLACE VIEW public.v\_waypoint\_attraction\_details\_localized AS

WITH attraction\_details\_translations AS (

SELECT

tr.row\_foreign\_key AS waypoint\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'detailed\_description', MAX(CASE WHEN tr.column\_identifier = 'detailed\_description' THEN tr.translated\_text ELSE NULL END),

'historical\_significance\_notes', MAX(CASE WHEN tr.column\_identifier = 'historical\_significance\_notes' THEN tr.translated\_text ELSE NULL END),

'cultural\_significance\_notes', MAX(CASE WHEN tr.column\_identifier = 'cultural\_significance\_notes' THEN tr.translated\_text ELSE NULL END),

'spiritual\_significance\_notes', MAX(CASE WHEN tr.column\_identifier = 'spiritual\_significance\_notes' THEN tr.translated\_text ELSE NULL END),

'key\_historical\_events\_notes', MAX(CASE WHEN tr.column\_identifier = 'key\_historical\_events\_notes' THEN tr.translated\_text ELSE NULL END),

'opening\_hours\_text\_notes', MAX(CASE WHEN tr.column\_identifier = 'opening\_hours\_text\_notes' THEN tr.translated\_text ELSE NULL END),

'entry\_fee\_details', MAX(CASE WHEN tr.column\_identifier = 'entry\_fee\_details' THEN tr.translated\_text ELSE NULL END),

'guided\_tours\_info', MAX(CASE WHEN tr.column\_identifier = 'guided\_tours\_info' THEN tr.translated\_text ELSE NULL END),

'audio\_guides\_info', MAX(CASE WHEN tr.column\_identifier = 'audio\_guides\_info' THEN tr.translated\_text ELSE NULL END),

'photography\_allowed\_notes', MAX(CASE WHEN tr.column\_identifier = 'photography\_allowed\_notes' THEN tr.translated\_text ELSE NULL END),

'accessibility\_details\_specific', MAX(CASE WHEN tr.column\_identifier = 'accessibility\_details\_specific' THEN tr.translated\_text ELSE NULL END)

-- Note: translations for 'associated\_historical\_figures\_text' (TEXT[]) elements are complex here and often handled at application layer or by transforming the array into related rows for translation.

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'attractions\_details'

GROUP BY tr.row\_foreign\_key

),

attraction\_type\_translations AS (

SELECT

tr.row\_foreign\_key AS type\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),

'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'attraction\_types\_master'

GROUP BY tr.row\_foreign\_key

),

visitor\_amenity\_translations AS (

SELECT

tr.row\_foreign\_key AS amenity\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),

'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'visitor\_amenities\_master'

GROUP BY tr.row\_foreign\_key

),

religious\_service\_type\_translations AS (

SELECT

tr.row\_foreign\_key AS type\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),

'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'religious\_service\_types\_master'

GROUP BY tr.row\_foreign\_key

),

religious\_service\_schedule\_translations AS (

SELECT

tr.row\_foreign\_key as schedule\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'schedule\_description\_text', MAX(CASE WHEN tr.column\_identifier = 'schedule\_description\_text' THEN tr.translated\_text ELSE NULL END),

'language\_notes', MAX(CASE WHEN tr.column\_identifier = 'language\_notes' THEN tr.translated\_text ELSE NULL END),

'location\_within\_site\_notes', MAX(CASE WHEN tr.column\_identifier = 'location\_within\_site\_notes' THEN tr.translated\_text ELSE NULL END),

'seasonal\_validity\_notes', MAX(CASE WHEN tr.column\_identifier = 'seasonal\_validity\_notes' THEN tr.translated\_text ELSE NULL END),

'service\_notes', MAX(CASE WHEN tr.column\_identifier = 'service\_notes' THEN tr.translated\_text ELSE NULL END)

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'religious\_service\_schedules'

GROUP BY tr.row\_foreign\_key

),

attraction\_media\_translations AS (

SELECT

tr.row\_foreign\_key AS media\_link\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'caption\_override', MAX(CASE WHEN tr.column\_identifier = 'caption\_override' THEN tr.translated\_text ELSE NULL END),

'alt\_text\_override', MAX(CASE WHEN tr.column\_identifier = 'alt\_text\_override' THEN tr.translated\_text ELSE NULL END)

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'attraction\_details\_media'

GROUP BY tr.row\_foreign\_key

)

SELECT

ad.waypoint\_id,

ad.attraction\_type\_id,

atm.code AS attraction\_type\_code,

atm.label AS attraction\_type\_label, -- English label

atm.icon\_identifier AS attraction\_type\_icon\_identifier,

COALESCE(att\_type\_tr.all\_translations, '{}'::jsonb) AS attraction\_type\_all\_translations,

ad.detailed\_description,

ad.historical\_significance\_notes,

ad.cultural\_significance\_notes,

ad.spiritual\_significance\_notes,

ad.associated\_historical\_figures\_text,

ad.key\_historical\_events\_notes,

ad.opening\_hours\_structured,

ad.opening\_hours\_text\_notes,

ad.opening\_hours\_last\_verified\_at,

ad.entry\_fee\_details,

ad.guided\_tours\_info,

ad.audio\_guides\_info,

ad.photography\_allowed\_notes,

ad.accessibility\_details\_specific,

COALESCE(ad\_tr.all\_translations, '{}'::jsonb) AS all\_translations,

(

SELECT jsonb\_agg(

jsonb\_build\_object(

'id', vam.id,

'code', vam.code,

'label', vam.label, -- English label

'icon\_identifier', vam.icon\_identifier,

'all\_translations', COALESCE(vam\_tr.all\_translations, '{}'::jsonb)

) ORDER BY vam.label -- Or some sort\_order if available

)

FROM unnest(ad.visitor\_amenity\_ids) amenity\_id

JOIN public.visitor\_amenities\_master vam ON vam.id = amenity\_id

LEFT JOIN visitor\_amenity\_translations vam\_tr ON vam\_tr.amenity\_id\_text = vam.id::TEXT

WHERE ad.visitor\_amenity\_ids IS NOT NULL

) AS visitor\_amenities,

(

SELECT jsonb\_agg(

jsonb\_build\_object(

'media\_id', adm\_link.media\_id,

'media\_role\_code', adm\_link.media\_role\_code,

'display\_order', adm\_link.display\_order,

'caption\_override', adm\_link.caption\_override, -- English

'alt\_text\_override', adm\_link.alt\_text\_override, -- English

'all\_translations', COALESCE(adm\_tr.all\_translations, '{}'::jsonb),

'image\_variants\_json', med.image\_variants\_json,

'original\_file\_name', med.file\_name\_original,

'original\_mime\_type', med.file\_mime\_type

) ORDER BY adm\_link.display\_order

)

FROM public.attraction\_details\_media adm\_link

JOIN public.media med ON med.id = adm\_link.media\_id

LEFT JOIN attraction\_media\_translations adm\_tr ON adm\_tr.media\_link\_id\_text = adm\_link.id::TEXT

WHERE adm\_link.attraction\_waypoint\_id = ad.waypoint\_id

) AS media\_gallery,

(

SELECT jsonb\_agg(

jsonb\_build\_object(

'id', rs.id,

'service\_type\_id', rs.service\_type\_id,

'service\_type\_code', rstm.code,

'service\_type\_label', rstm.label, -- English

'service\_type\_icon\_identifier', rstm.icon\_identifier,

'service\_type\_all\_translations', COALESCE(rst\_tr.all\_translations, '{}'::jsonb),

'schedule\_description\_text', rs.schedule\_description\_text, -- English

'days\_of\_week', rs.days\_of\_week,

'time\_of\_day', rs.time\_of\_day,

'language\_code', rs.language\_code,

'language\_name\_en', lm.display\_name\_en, -- Assumes languages\_master has display\_name\_en

'language\_all\_translations', jsonb\_build\_object(lm.language\_code, jsonb\_build\_object('name', lm.display\_name\_native)), -- Simplified, needs full lang translations

'language\_notes', rs.language\_notes, -- English

'location\_within\_site\_notes', rs.location\_within\_site\_notes, -- English

'seasonal\_validity\_notes', rs.seasonal\_validity\_notes, -- English

'is\_pilgrim\_specific\_service', rs.is\_pilgrim\_specific\_service,

'service\_notes', rs.service\_notes, -- English

'all\_translations', COALESCE(rs\_tr.all\_translations, '{}'::jsonb),

'data\_last\_verified\_at', rs.data\_last\_verified\_at

) ORDER BY rstm.sort\_order, rs.time\_of\_day NULLS FIRST

)

FROM public.religious\_service\_schedules rs

JOIN public.religious\_service\_types\_master rstm ON rstm.id = rs.service\_type\_id

LEFT JOIN public.languages\_master lm ON lm.language\_code = rs.language\_code

LEFT JOIN religious\_service\_type\_translations rst\_tr ON rst\_tr.type\_id\_text = rstm.id::TEXT

LEFT JOIN religious\_service\_schedule\_translations rs\_tr ON rs\_tr.schedule\_id\_text = rs.id::TEXT

WHERE rs.attraction\_waypoint\_id = ad.waypoint\_id

) AS religious\_services,

ad.data\_last\_verified\_at,

ad.created\_at,

creator\_profile.full\_name AS created\_by\_full\_name,

ad.updated\_at,

updater\_profile.full\_name AS updated\_by\_full\_name,

wp.name AS parent\_waypoint\_name, -- English name from waypoints

wp.slug AS parent\_waypoint\_slug

FROM

public.attractions\_details ad

JOIN

public.waypoints wp ON ad.waypoint\_id = wp.id -- For parent waypoint info & RLS context

JOIN

public.attraction\_types\_master atm ON ad.attraction\_type\_id = atm.id

LEFT JOIN

attraction\_details\_translations ad\_tr ON ad\_tr.waypoint\_id\_text = ad.waypoint\_id::TEXT

LEFT JOIN

attraction\_type\_translations att\_type\_tr ON att\_type\_tr.type\_id\_text = atm.id::TEXT

LEFT JOIN

public.profiles creator\_profile ON ad.created\_by\_profile\_id = creator\_profile.id

LEFT JOIN

public.profiles updater\_profile ON ad.updated\_by\_profile\_id = updater\_profile.id;

COMMENT ON VIEW public.v\_waypoint\_attraction\_details\_localized IS 'Provides a denormalized and localized view of attraction details, including types, amenities, media, and religious service schedules. Version 1.0';

```

\* \* \* \* \*

5\. Key Dependencies

- `public.attractions\_details` (V1.3)

- `public.waypoints` (V1.3+ assumed for fields like `name`, `slug`, `content\_visibility\_status\_id`, `deleted\_at`)

- `public.attraction\_types\_master` (V1.1)

- `public.visitor\_amenities\_master` (V1.1)

- `public.religious\_service\_schedules` (V1.3)

- `public.religious\_service\_types\_master` (V1.1)

- `public.languages\_master` (V2.1)

- `public.media` (V2.2)

- `public.attraction\_details\_media` (V1.0)

- `public.media\_roles\_master`

- `public.translations` (V2.1)

- `public.profiles` (for `created\_by`/`updated\_by` names)

- `public.content\_statuses\_master` (for RLS via `waypoints`)

\* \* \* \* \*

6\. Performance Considerations

- Complex Joins: The view involves numerous joins and subqueries, especially for aggregating translations and nested JSON arrays (amenities, media, services). This can be resource-intensive.

- Translation Aggregation: The CTEs for `\*\_translations` use `jsonb\_object\_agg` and `GROUP BY`, which can be costly on a large `translations` table. Ensure `translations.table\_identifier`, `translations.column\_identifier`, and `translations.row\_foreign\_key` are well-indexed. The existing `idx\_translations\_lookup` is good.

- Array Expansion: Unnesting `visitor\_amenity\_ids` and then joining to `visitor\_amenities\_master` and its translations for each attraction can be intensive if there are many amenities per attraction or many attractions queried at once.

- JSONB Aggregation: Using `jsonb\_agg` for `visitor\_amenities`, `media\_gallery`, and `religious\_services` adds overhead.

- Indexes on Base Tables:

- All foreign key columns used in joins in the base tables must be indexed (e.g., `attractions\_details.attraction\_type\_id`, `religious\_service\_schedules.service\_type\_id`, etc.).

- The `row\_foreign\_key` in `public.translations` should be indexed.

- `waypoints.id` (PK), `attractions\_details.waypoint\_id` (PK).

- Materialized View: If read performance for this aggregated data becomes critical and the underlying data doesn't change with extremely high frequency, converting this view to a MATERIALIZED VIEW (refreshed periodically) would significantly improve query speed for the API. The refresh strategy (e.g., on commit, nightly) would need careful consideration.

- Filtering: If this view is queried with `WHERE` clauses on fields from `attractions\_details` or `waypoints`, ensure those base table columns are indexed.

\* \* \* \* \*

7\. RLS & Security Notes

- The view itself does not have RLS policies applied directly. Access to data through this view is governed by the RLS policies on the underlying base tables, primarily `public.attractions\_details` and `public.waypoints`.

- A user querying this view will only see rows for attractions they are permitted to see based on the `SELECT` policies on `attractions\_details`. The `attractions\_details` RLS policy, in turn, checks the publication status and soft-deletion status of the parent `waypoints` record.

- The `SECURITY DEFINER` option might be necessary for the translation CTE functions if direct access to `public.translations` is restricted for the querying role but the view needs to aggregate this data. However, it's generally safer for views to operate with `SECURITY INVOKER` (default) if RLS on base tables allows sufficient access. The CTEs in the provided SQL are part of the main view query and will run with the view invoker's permissions.

\* \* \* \* \*

8\. API Endpoints Supported

- Primarily supports: `/waypoints/{waypoint\_id}/attraction\_details` (GET).

- The API backend would query this view using `WHERE waypoint\_id = {input\_id}`.

- The `lang` query parameter would be handled by the API backend to select the appropriate translated text from the `all\_translations` JSONB objects within the view's results or by using the primary language fields.

\* \* \* \* \*

9\. Rationale for Creation

- Simplification: Drastically simplifies query logic in the API backend by providing a pre-joined, structured, and localized dataset for attractions.

- Consistency: Ensures a consistent data structure is available for attraction details across different parts of the application.

- Maintainability: Centralizes the complex data aggregation and localization logic in the database layer, making the application code cleaner.

- Performance (Potential): While complex, a well-indexed view (or materialized view) can be more performant than having the application execute multiple discrete queries and assemble the data itself.

------

View Specification: `public.v\_waypoint\_food\_water\_source\_details\_localized`

\* \* \* \* \*

1\. View Name

`public.v\_waypoint\_food\_water\_source\_details\_localized`

\* \* \* \* \*

2\. Purpose & Primary Use-Cases

- Purpose: To provide a comprehensive, denormalized, and localized representation of a food or water source's details, including its type, potability, reliability, operational notes, associated media, and any commercial aspects like price range or meal types.

- Primary Use-Cases:

- Power the API endpoint `/waypoints/{waypoint\_id}/food\_water\_source\_details` by simplifying backend query logic.

- Provide all necessary data for displaying a complete food/water source detail page in a web or mobile application, with support for internationalization.

- Enable users to easily understand the nature, reliability, and specifics of a food or water point.

\* \* \* \* \*

3\. View Schema (Columns)

\*(This view will select from `food\_water\_sources\_details` and join various other tables. Referenced master data will have English labels directly included and their full translation sets also available within nested JSON structures.)\*

| Column Name | Data Type | Description |

| `waypoint\_id` | `BIGINT` | Inherited from `food\_water\_sources\_details`. Links to `public.waypoints(id)`. PK of the detail record. |

| `source\_type\_id` | `INTEGER` | Inherited from `food\_water\_sources\_details`. FK to `public.food\_water\_source\_types\_master(id)`. |

| `source\_type\_code` | `TEXT` | From `food\_water\_source\_types\_master.code`. |

| `source\_type\_label` | `TEXT` | Primary reference language (English) label from `food\_water\_source\_types\_master.label`. |

| `source\_type\_is\_commercial` | `BOOLEAN` | From `food\_water\_source\_types\_master.is\_commercial`. |

| `source\_type\_icon\_identifier` | `TEXT` | From `food\_water\_source\_types\_master.icon\_identifier`. |

| `source\_type\_all\_translations` | `JSONB` | All translations for the `label` and `description` of the referenced `food\_water\_source\_types\_master` record. |

| `is\_potable\_water\_source` | `BOOLEAN` | Inherited from `food\_water\_sources\_details.is\_potable\_water\_source`. |

| `water\_reliability\_id` | `INTEGER` | Inherited from `food\_water\_sources\_details`. FK to `public.water\_reliability\_types\_master(id)`. |

| `water\_reliability\_code` | `TEXT` | From `water\_reliability\_types\_master.code`. |

| `water\_reliability\_label` | `TEXT` | Primary reference language (English) label from `water\_reliability\_types\_master.label`. |

| `water\_reliability\_icon\_identifier` | `TEXT` | From `water\_reliability\_types\_master.icon\_identifier`. |

| `water\_reliability\_advisory\_level` | `SMALLINT` | From `water\_reliability\_types\_master.advisory\_level`. |

| `water\_reliability\_all\_translations` | `JSONB` | All translations for the `label` and `description` of the referenced `water\_reliability\_types\_master` record. |

| `water\_source\_access\_notes` | `TEXT` | Primary reference language (English) text from `food\_water\_sources\_details.water\_source\_access\_notes`. |

| `establishment\_price\_range\_id` | `INTEGER` | Inherited from `food\_water\_sources\_details`. FK to `public.establishment\_price\_ranges\_master(id)`. |

| `establishment\_price\_range\_code` | `TEXT` | From `establishment\_price\_ranges\_master.code`. |

| `establishment\_price\_range\_label` | `TEXT` | Primary reference language (English) label from `establishment\_price\_ranges\_master.label`. |

| `establishment\_price\_range\_symbol` | `TEXT` | From `establishment\_price\_ranges\_master.symbol`. |

| `establishment\_price\_range\_all\_translations` | `JSONB` | All translations for the `label` and `description` of the referenced `establishment\_price\_ranges\_master` record. |

| `highlighted\_dishes\_local\_specialties` | `TEXT[]` | Inherited from `food\_water\_sources\_details`. Elements are primary reference language (English). (Individual elements translatable via `public.translations` if keyed appropriately). |

| `opening\_hours\_structured` | `JSONB` | Inherited from `food\_water\_sources\_details.opening\_hours\_structured`. |

| `opening\_hours\_text\_notes` | `TEXT` | Primary reference language (English) text from `food\_water\_sources\_details.opening\_hours\_text\_notes`. |

| `opening\_hours\_last\_verified\_at` | `TIMESTAMPTZ` | Inherited from `food\_water\_sources\_details.opening\_hours\_last\_verified\_at`. |

| `outdoor\_seating\_available` | `BOOLEAN` | Inherited from `food\_water\_sources\_details.outdoor\_seating\_available`. |

| `specific\_notes\_for\_pilgrims` | `TEXT` | Primary reference language (English) text from `food\_water\_sources\_details.specific\_notes\_for\_pilgrims`. |

| `all\_translations` | `JSONB` | All translations for translatable fields directly on `food\_water\_sources\_details` (e.g., `water\_source\_access\_notes`, etc.), keyed by language code and then field name. |

| `serves\_meal\_types` | `JSONB` | Array of JSON objects, each representing a meal type. Includes `id`, `code`, `label` (English), `icon\_identifier`, and `all\_translations` for the meal type's label. |

| `dietary\_options` | `JSONB` | Array of JSON objects, each representing a dietary option. Includes `id`, `code`, `label` (English), `icon\_identifier`, and `all\_translations` for the option's label/description. |

| `payment\_methods` | `JSONB` | Array of JSON objects, each representing a payment method. Includes `id`, `code`, `label` (English), `icon\_identifier`, and `all\_translations` for the method's label/description. |

| `media\_gallery` | `JSONB` | Array of JSON objects, each representing a media item. Includes `media\_id`, `media\_role\_code`, `alt\_text` (English), `caption` (English), `image\_variants\_json`, and `all\_translations` for alt/caption. |

| `data\_last\_verified\_at` | `TIMESTAMPTZ` | Inherited from `food\_water\_sources\_details.data\_last\_verified\_at`. |

| `created\_at` | `TIMESTAMPTZ` | Inherited from `food\_water\_sources\_details.created\_at`. |

| `created\_by\_full\_name` | `TEXT` | Full name of the profile that created the record (joined from `public.profiles`). |

| `updated\_at` | `TIMESTAMPTZ` | Inherited from `food\_water\_sources\_details.updated\_at`. |

| `updated\_by\_full\_name` | `TEXT` | Full name of the profile that last updated the record (joined from `public.profiles`). |

| `parent\_waypoint\_name` | `TEXT` | The `name` of the parent waypoint from `public.waypoints` (primary language). |

| `parent\_waypoint\_slug` | `TEXT` | The `slug` of the parent waypoint from `public.waypoints`. |

\* \* \* \* \*

4\. Underlying SQL Definition

SQL

```

CREATE OR REPLACE VIEW public.v\_waypoint\_food\_water\_source\_details\_localized AS

WITH direct\_translations AS (

SELECT

tr.row\_foreign\_key AS waypoint\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'water\_source\_access\_notes', MAX(CASE WHEN tr.column\_identifier = 'water\_source\_access\_notes' THEN tr.translated\_text ELSE NULL END),

'opening\_hours\_text\_notes', MAX(CASE WHEN tr.column\_identifier = 'opening\_hours\_text\_notes' THEN tr.translated\_text ELSE NULL END),

'specific\_notes\_for\_pilgrims', MAX(CASE WHEN tr.column\_identifier = 'specific\_notes\_for\_pilgrims' THEN tr.translated\_text ELSE NULL END)

-- highlighted\_dishes\_local\_specialties TEXT[] elements are translated individually if needed, complex for this CTE.

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'food\_water\_sources\_details'

GROUP BY tr.row\_foreign\_key

),

source\_type\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'food\_water\_source\_types\_master' GROUP BY tr.row\_foreign\_key

),

reliability\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'water\_reliability\_types\_master' GROUP BY tr.row\_foreign\_key

),

price\_range\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'establishment\_price\_ranges\_master' GROUP BY tr.row\_foreign\_key

),

meal\_type\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'meal\_type\_tags\_master' GROUP BY tr.row\_foreign\_key

),

dietary\_option\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'dietary\_option\_tags\_master' GROUP BY tr.row\_foreign\_key

),

payment\_method\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'payment\_methods\_master' GROUP BY tr.row\_foreign\_key

),

fw\_media\_translations AS (

SELECT tr.row\_foreign\_key AS media\_link\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('caption\_override', MAX(CASE WHEN tr.column\_identifier = 'caption\_override' THEN tr.translated\_text ELSE NULL END), 'alt\_text\_override', MAX(CASE WHEN tr.column\_identifier = 'alt\_text\_override' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'food\_water\_source\_media' GROUP BY tr.row\_foreign\_key

)

SELECT

fwsd.waypoint\_id,

fwsd.source\_type\_id,

fstm.code AS source\_type\_code,

fstm.label AS source\_type\_label, -- English

fstm.is\_commercial AS source\_type\_is\_commercial,

fstm.icon\_identifier AS source\_type\_icon\_identifier,

COALESCE(st\_tr.all\_translations, '{}'::jsonb) AS source\_type\_all\_translations,

fwsd.is\_potable\_water\_source,

fwsd.water\_reliability\_id,

wrtm.code AS water\_reliability\_code,

wrtm.label AS water\_reliability\_label, -- English

wrtm.icon\_identifier AS water\_reliability\_icon\_identifier,

wrtm.advisory\_level AS water\_reliability\_advisory\_level,

COALESCE(wr\_tr.all\_translations, '{}'::jsonb) AS water\_reliability\_all\_translations,

fwsd.water\_source\_access\_notes, -- English

fwsd.establishment\_price\_range\_id,

eprm.code AS establishment\_price\_range\_code,

eprm.label AS establishment\_price\_range\_label, -- English

eprm.symbol AS establishment\_price\_range\_symbol,

COALESCE(epr\_tr.all\_translations, '{}'::jsonb) AS establishment\_price\_range\_all\_translations,

fwsd.highlighted\_dishes\_local\_specialties, -- English array

fwsd.opening\_hours\_structured,

fwsd.opening\_hours\_text\_notes, -- English

fwsd.opening\_hours\_last\_verified\_at,

fwsd.outdoor\_seating\_available,

fwsd.specific\_notes\_for\_pilgrims, -- English

COALESCE(fwsd\_tr.all\_translations, '{}'::jsonb) AS all\_translations,

(SELECT jsonb\_agg(jsonb\_build\_object('id', mttm.id, 'code', mttm.code, 'label', mttm.label, 'icon\_identifier', mttm.icon\_identifier, 'all\_translations', COALESCE(mt\_tr.all\_translations, '{}'::jsonb))) FROM unnest(fwsd.serves\_meal\_type\_tag\_ids) tag\_id JOIN public.meal\_type\_tags\_master mttm ON mttm.id = tag\_id LEFT JOIN meal\_type\_translations mt\_tr ON mt\_tr.master\_id\_text = mttm.id::TEXT WHERE fwsd.serves\_meal\_type\_tag\_ids IS NOT NULL) AS serves\_meal\_types,

(SELECT jsonb\_agg(jsonb\_build\_object('id', dotm.id, 'code', dotm.code, 'label', dotm.label, 'icon\_identifier', dotm.icon\_identifier, 'all\_translations', COALESCE(do\_tr.all\_translations, '{}'::jsonb))) FROM unnest(fwsd.dietary\_option\_tag\_ids) tag\_id JOIN public.dietary\_option\_tags\_master dotm ON dotm.id = tag\_id LEFT JOIN dietary\_option\_translations do\_tr ON do\_tr.master\_id\_text = dotm.id::TEXT WHERE fwsd.dietary\_option\_tag\_ids IS NOT NULL) AS dietary\_options,

(SELECT jsonb\_agg(jsonb\_build\_object('id', pmm.id, 'code', pmm.code, 'label', pmm.label, 'icon\_identifier', pmm.icon\_identifier, 'all\_translations', COALESCE(pm\_tr.all\_translations, '{}'::jsonb))) FROM unnest(fwsd.payment\_method\_tag\_ids) tag\_id JOIN public.payment\_methods\_master pmm ON pmm.id = tag\_id LEFT JOIN payment\_method\_translations pm\_tr ON pm\_tr.master\_id\_text = pmm.id::TEXT WHERE fwsd.payment\_method\_tag\_ids IS NOT NULL) AS payment\_methods,

(SELECT jsonb\_agg(jsonb\_build\_object('media\_id', fwm\_link.media\_id, 'media\_role\_code', fwm\_link.media\_role\_code, 'display\_order', fwm\_link.display\_order, 'caption\_override', fwm\_link.caption\_override, 'alt\_text\_override', fwm\_link.alt\_text\_override, 'all\_translations', COALESCE(fwm\_tr.all\_translations, '{}'::jsonb),'image\_variants\_json', med.image\_variants\_json, 'original\_file\_name', med.file\_name\_original, 'original\_mime\_type', med.file\_mime\_type) ORDER BY fwm\_link.display\_order) FROM public.food\_water\_source\_media fwm\_link JOIN public.media med ON med.id = fwm\_link.media\_id LEFT JOIN fw\_media\_translations fwm\_tr ON fwm\_tr.media\_link\_id\_text = fwm\_link.id::TEXT WHERE fwm\_link.food\_water\_source\_waypoint\_id = fwsd.waypoint\_id) AS media\_gallery,

fwsd.data\_last\_verified\_at,

fwsd.created\_at,

creator\_profile.full\_name AS created\_by\_full\_name,

fwsd.updated\_at,

updater\_profile.full\_name AS updated\_by\_full\_name,

wp.name AS parent\_waypoint\_name, -- English name from waypoints

wp.slug AS parent\_waypoint\_slug

FROM

public.food\_water\_sources\_details fwsd

JOIN

public.waypoints wp ON fwsd.waypoint\_id = wp.id

JOIN

public.food\_water\_source\_types\_master fstm ON fwsd.source\_type\_id = fstm.id

LEFT JOIN

public.water\_reliability\_types\_master wrtm ON fwsd.water\_reliability\_id = wrtm.id

LEFT JOIN

public.establishment\_price\_ranges\_master eprm ON fwsd.establishment\_price\_range\_id = eprm.id

LEFT JOIN

direct\_translations fwsd\_tr ON fwsd\_tr.waypoint\_id\_text = fwsd.waypoint\_id::TEXT

LEFT JOIN

source\_type\_translations st\_tr ON st\_tr.master\_id\_text = fstm.id::TEXT

LEFT JOIN

reliability\_translations wr\_tr ON wr\_tr.master\_id\_text = wrtm.id::TEXT

LEFT JOIN

price\_range\_translations epr\_tr ON epr\_tr.master\_id\_text = eprm.id::TEXT

LEFT JOIN

public.profiles creator\_profile ON fwsd.created\_by\_profile\_id = creator\_profile.id

LEFT JOIN

public.profiles updater\_profile ON fwsd.updated\_by\_profile\_id = updater\_profile.id;

COMMENT ON VIEW public.v\_waypoint\_food\_water\_source\_details\_localized IS 'Provides a denormalized and localized view of food and water source details, including types, reliability, commercial aspects, and media. Version 1.0';

```

\* \* \* \* \*

5\. Key Dependencies

- `public.food\_water\_sources\_details` (V1.3)

- `public.waypoints` (V1.3+ assumed)

- `public.food\_water\_source\_types\_master` (V1.2)

- `public.water\_reliability\_types\_master` (V1.2)

- `public.establishment\_price\_ranges\_master` (V1.1)

- `public.meal\_type\_tags\_master` (V1.1)

- `public.dietary\_option\_tags\_master` (V1.1)

- `public.payment\_methods\_master` (V1.1)

- `public.media` (V2.2)

- `public.food\_water\_source\_media` (V1.0)

- `public.media\_roles\_master`

- `public.translations` (V2.1)

- `public.profiles`

- `public.content\_statuses\_master`

\* \* \* \* \*

6\. Performance Considerations

- Highly Complex: This view is very complex due to numerous joins, multiple CTEs for translations of various entities, and multiple `jsonb\_agg` subqueries for array FKs and linked media. Performance will be a significant concern if queried frequently without optimization or on very large datasets.

- Translation Aggregation: Multiple CTEs aggregate translations. The `translations` table must have excellent indexing on `(table\_identifier, row\_foreign\_key, language\_code, column\_identifier)`.

- Array FK Expansion: Unnesting `serves\_meal\_type\_tag\_ids`, `dietary\_option\_tag\_ids`, and `payment\_method\_tag\_ids`, then joining to their respective master tables and translation CTEs for each food/water source, is resource-intensive.

- Indexes on Base Tables: All foreign keys in the base tables (`food\_water\_sources\_details`, master tables) involved in joins must be indexed. Primary keys are inherently indexed. GIN indexes on array columns in `food\_water\_sources\_details` help if filtering by tags, but here we are expanding them.

- Materialized View Recommendation: Given the complexity, if this view is frequently accessed by the API for read operations and can tolerate some data latency, converting it to a MATERIALIZED VIEW is highly recommended. A refresh strategy (e.g., nightly, or on demand after significant data changes) would be necessary.

- Querying the View: When querying this view, filtering by `waypoint\_id` will be efficient as it's the PK of the main detail table.

\* \* \* \* \*

7\. RLS & Security Notes

- The view operates under the permissions of the querying user (`SECURITY INVOKER` by default).

- Data visibility is primarily controlled by RLS policies on the underlying `public.food\_water\_sources\_details` and `public.waypoints` tables. The view will only return rows that the user is permitted to see based on these policies (e.g., details for published and non-deleted waypoints).

- CTEs accessing `public.translations` will also respect any RLS on that table, though typically translations are readable if the parent record is readable.

\* \* \* \* \*

8\. API Endpoints Supported

- Primarily supports: `/waypoints/{waypoint\_id}/food\_water\_source\_details` (GET).

- The API backend would query this view: `SELECT \* FROM public.v\_waypoint\_food\_water\_source\_details\_localized WHERE waypoint\_id = {input\_id};`.

- The `lang` query parameter from the API request would be used by the API backend to extract the relevant language versions from the `all\_translations` JSONB objects or use the direct primary language fields (e.g., `source\_type\_label`).

\* \* \* \* \*

9\. Rationale for Creation

- API Query Simplification: Abstract away the numerous joins and complex data aggregation (especially for translations and array-based relationships) required to construct a full food/water source detail object.

- Data Consistency for API: Provides a single, consistent source of truth for the API to fetch comprehensive food/water source details.

- Improved Maintainability: Centralizes data shaping logic in the database layer, making the API backend code cleaner and easier to maintain. Changes to underlying table structures or join logic can often be handled within the view without altering the API's query to the view.

-----

View Specification: `public.v\_waypoint\_shop\_service\_details\_localized`

\* \* \* \* \*

1\. View Name

`public.v\_waypoint\_shop\_service\_details\_localized`

\* \* \* \* \*

2\. Purpose & Primary Use-Cases

- Purpose: To provide a comprehensive, denormalized, and localized representation of a shop or practical service's details, including its type, operational specifics like opening hours, contact information, languages spoken, payment methods, associated media, and any pilgrim-relevant notes.

- Primary Use-Cases:

- Power the API endpoint `/waypoints/{waypoint\_id}/shop\_service\_details` by simplifying backend query logic.

- Furnish all necessary data for displaying a complete shop/service detail page in a web or mobile application, with robust internationalization support.

- Enable users to easily find and understand the specifics of essential services available to them.

\* \* \* \* \*

3\. View Schema (Columns)

\*(This view will select from `shops\_and\_services\_details` and join various other tables. Referenced master data will have English labels directly included, and their full translation sets will also be available within nested JSON structures.)\*

| Column Name | Data Type | Description |

| `waypoint\_id` | `BIGINT` | Inherited from `shops\_and\_services\_details`. Links to `public.waypoints(id)`. PK of the detail record. |

| `service\_type\_id` | `INTEGER` | Inherited from `shops\_and\_services\_details`. FK to `public.shop\_service\_types\_master(id)`. |

| `service\_type\_code` | `TEXT` | From `shop\_service\_types\_master.code`. |

| `service\_type\_label` | `TEXT` | Primary reference language (English) label from `shop\_service\_types\_master.label`. |

| `service\_type\_category` | `TEXT` | From `shop\_service\_types\_master.category`. |

| `service\_type\_icon\_identifier` | `TEXT` | From `shop\_service\_types\_master.icon\_identifier`. |

| `service\_type\_all\_translations` | `JSONB` | All translations for the `label` and `description` of the referenced `shop\_service\_types\_master` record. |

| `operator\_or\_brand\_name` | `TEXT` | Primary reference language (English) text from `shops\_and\_services\_details.operator\_or\_brand\_name`. |

| `products\_or\_services\_summary` | `TEXT` | Primary reference language (English) text from `shops\_and\_services\_details.products\_or\_services\_summary`. |

| `is\_open\_24\_hours` | `BOOLEAN` | Inherited from `shops\_and\_services\_details.is\_open\_24\_hours`. |

| `opening\_hours\_structured` | `JSONB` | Inherited from `shops\_and\_services\_details.opening\_hours\_structured`. |

| `opening\_hours\_text\_notes` | `TEXT` | Primary reference language (English) text from `shops\_and\_services\_details.opening\_hours\_text\_notes`. |

| `opening\_hours\_last\_verified\_at` | `TIMESTAMPTZ` | Inherited from `shops\_and\_services\_details.opening\_hours\_last\_verified\_at`. |

| `emergency\_service\_details` | `TEXT` | Primary reference language (English) text from `shops\_and\_services\_details.emergency\_service\_details`. |

| `contact\_phone\_service` | `TEXT` | Inherited from `shops\_and\_services\_details.contact\_phone\_service`. |

| `contact\_email\_service` | `TEXT` | Inherited from `shops\_and\_services\_details.contact\_email\_service`. |

| `website\_url\_service` | `TEXT` | Inherited from `shops\_and\_services\_details.website\_url\_service`. |

| `general\_price\_range\_id` | `INTEGER` | Inherited from `shops\_and\_services\_details`. FK to `public.establishment\_price\_ranges\_master(id)`. |

| `general\_price\_range\_code` | `TEXT` | From `establishment\_price\_ranges\_master.code`. |

| `general\_price\_range\_label` | `TEXT` | Primary reference language (English) label from `establishment\_price\_ranges\_master.label`. |

| `general\_price\_range\_symbol` | `TEXT` | From `establishment\_price\_ranges\_master.symbol`. |

| `general\_price\_range\_all\_translations` | `JSONB` | All translations for the `label` and `description` of the referenced `establishment\_price\_ranges\_master` record. |

| `accessibility\_notes\_service` | `TEXT` | Primary reference language (English) text from `shops\_and\_services\_details.accessibility\_notes\_service`. |

| `service\_specific\_details\_notes` | `TEXT` | Primary reference language (English) text from `shops\_and\_services\_details.service\_specific\_details\_notes`. |

| `all\_translations` | `JSONB` | All translations for translatable fields directly on `shops\_and\_services\_details` (e.g., `operator\_or\_brand\_name`, etc.), keyed by language code and then field name. |

| `languages\_spoken` | `JSONB` | Array of JSON objects, each representing a language spoken. Includes `language\_code`, `name\_en` (English name from `languages\_master`), `name\_native`, and `all\_translations` for the language names. |

| `payment\_methods` | `JSONB` | Array of JSON objects, each representing an accepted payment method. Includes `id`, `code`, `label` (English), `icon\_identifier`, and `all\_translations` for the method's label/description. |

| `media\_gallery` | `JSONB` | Array of JSON objects, each representing a media item. Includes `media\_id`, `media\_role\_code`, `alt\_text` (English), `caption` (English), `image\_variants\_json`, and `all\_translations` for alt/caption. |

| `data\_last\_verified\_at` | `TIMESTAMPTZ` | Inherited from `shops\_and\_services\_details.data\_last\_verified\_at`. |

| `deleted\_at` | `TIMESTAMPTZ` | Inherited from `shops\_and\_services\_details.deleted\_at`. Soft deletion timestamp for this specific service detail. |

| `created\_at` | `TIMESTAMPTZ` | Inherited from `shops\_and\_services\_details.created\_at`. |

| `created\_by\_full\_name` | `TEXT` | Full name of the profile that created the record (joined from `public.profiles`). |

| `updated\_at` | `TIMESTAMPTZ` | Inherited from `shops\_and\_services\_details.updated\_at`. |

| `updated\_by\_full\_name` | `TEXT` | Full name of the profile that last updated the record (joined from `public.profiles`). |

| `parent\_waypoint\_name` | `TEXT` | The `name` of the parent waypoint from `public.waypoints` (primary language). |

| `parent\_waypoint\_slug` | `TEXT` | The `slug` of the parent waypoint from `public.waypoints`. |

\* \* \* \* \*

4\. Underlying SQL Definition

SQL

```

CREATE OR REPLACE VIEW public.v\_waypoint\_shop\_service\_details\_localized AS

WITH direct\_translations AS (

SELECT

tr.row\_foreign\_key AS waypoint\_id\_text,

jsonb\_object\_agg(

tr.language\_code,

jsonb\_strip\_nulls(jsonb\_build\_object(

'operator\_or\_brand\_name', MAX(CASE WHEN tr.column\_identifier = 'operator\_or\_brand\_name' THEN tr.translated\_text ELSE NULL END),

'products\_or\_services\_summary', MAX(CASE WHEN tr.column\_identifier = 'products\_or\_services\_summary' THEN tr.translated\_text ELSE NULL END),

'opening\_hours\_text\_notes', MAX(CASE WHEN tr.column\_identifier = 'opening\_hours\_text\_notes' THEN tr.translated\_text ELSE NULL END),

'emergency\_service\_details', MAX(CASE WHEN tr.column\_identifier = 'emergency\_service\_details' THEN tr.translated\_text ELSE NULL END),

'accessibility\_notes\_service', MAX(CASE WHEN tr.column\_identifier = 'accessibility\_notes\_service' THEN tr.translated\_text ELSE NULL END),

'service\_specific\_details\_notes', MAX(CASE WHEN tr.column\_identifier = 'service\_specific\_details\_notes' THEN tr.translated\_text ELSE NULL END)

))

) AS all\_translations

FROM public.translations tr

WHERE tr.table\_identifier = 'shops\_and\_services\_details'

GROUP BY tr.row\_foreign\_key

),

service\_type\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'shop\_service\_types\_master' GROUP BY tr.row\_foreign\_key

),

price\_range\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END),'description', MAX(CASE WHEN tr.column\_identifier = 'description' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'establishment\_price\_ranges\_master' GROUP BY tr.row\_foreign\_key

),

language\_master\_translations AS ( -- For translating language names themselves

SELECT tr.row\_foreign\_key AS lang\_code\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('display\_name', MAX(CASE WHEN tr.column\_identifier = 'display\_name\_native' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations -- Example, could be display\_name\_en if that's what gets translated

FROM public.translations tr WHERE tr.table\_identifier = 'languages\_master' GROUP BY tr.row\_foreign\_key

),

payment\_method\_translations AS (

SELECT tr.row\_foreign\_key AS master\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('label', MAX(CASE WHEN tr.column\_identifier = 'label' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'payment\_methods\_master' GROUP BY tr.row\_foreign\_key

),

ss\_media\_translations AS (

SELECT tr.row\_foreign\_key AS media\_link\_id\_text, jsonb\_object\_agg(tr.language\_code, jsonb\_strip\_nulls(jsonb\_build\_object('caption\_override', MAX(CASE WHEN tr.column\_identifier = 'caption\_override' THEN tr.translated\_text ELSE NULL END), 'alt\_text\_override', MAX(CASE WHEN tr.column\_identifier = 'alt\_text\_override' THEN tr.translated\_text ELSE NULL END)))) AS all\_translations

FROM public.translations tr WHERE tr.table\_identifier = 'shop\_service\_media' GROUP BY tr.row\_foreign\_key

)

SELECT

sssd.waypoint\_id,

sssd.service\_type\_id,

sstm.code AS service\_type\_code,

sstm.label AS service\_type\_label, -- English

sstm.category AS service\_type\_category,

sstm.icon\_identifier AS service\_type\_icon\_identifier,

COALESCE(st\_tr.all\_translations, '{}'::jsonb) AS service\_type\_all\_translations,

sssd.operator\_or\_brand\_name, -- English

sssd.products\_or\_services\_summary, -- English

sssd.is\_open\_24\_hours,

sssd.opening\_hours\_structured,

sssd.opening\_hours\_text\_notes, -- English

sssd.opening\_hours\_last\_verified\_at,

sssd.emergency\_service\_details, -- English

sssd.contact\_phone\_service,

sssd.contact\_email\_service,

sssd.website\_url\_service,

sssd.general\_price\_range\_id,

eprm.code AS general\_price\_range\_code,

eprm.label AS general\_price\_range\_label, -- English

eprm.symbol AS general\_price\_range\_symbol,

COALESCE(epr\_tr.all\_translations, '{}'::jsonb) AS general\_price\_range\_all\_translations,

sssd.accessibility\_notes\_service, -- English

sssd.service\_specific\_details\_notes, -- English

COALESCE(sssd\_tr.all\_translations, '{}'::jsonb) AS all\_translations,

(SELECT jsonb\_agg(jsonb\_build\_object('code', lm.language\_code, 'name\_en', lm.display\_name\_en, 'name\_native', lm.display\_name\_native, 'icon\_identifier', lm.icon\_identifier, 'all\_translations', COALESCE(lang\_m\_tr.all\_translations, '{}'::jsonb))) FROM unnest(sssd.languages\_spoken\_codes) lang\_c JOIN public.languages\_master lm ON lm.language\_code = lang\_c LEFT JOIN language\_master\_translations lang\_m\_tr ON lang\_m\_tr.lang\_code\_text = lm.language\_code WHERE sssd.languages\_spoken\_codes IS NOT NULL) AS languages\_spoken,

(SELECT jsonb\_agg(jsonb\_build\_object('id', pmm.id, 'code', pmm.code, 'label', pmm.label, 'icon\_identifier', pmm.icon\_identifier, 'all\_translations', COALESCE(pm\_tr.all\_translations, '{}'::jsonb))) FROM unnest(sssd.payment\_method\_ids) p\_id JOIN public.payment\_methods\_master pmm ON pmm.id = p\_id LEFT JOIN payment\_method\_translations pm\_tr ON pm\_tr.master\_id\_text = pmm.id::TEXT WHERE sssd.payment\_method\_ids IS NOT NULL) AS payment\_methods,

(SELECT jsonb\_agg(jsonb\_build\_object('media\_id', ssm\_link.media\_id, 'media\_role\_code', ssm\_link.media\_role\_code, 'display\_order', ssm\_link.display\_order, 'caption\_override', ssm\_link.caption\_override, 'alt\_text\_override', ssm\_link.alt\_text\_override, 'all\_translations', COALESCE(ssm\_tr.all\_translations, '{}'::jsonb),'image\_variants\_json', med.image\_variants\_json, 'original\_file\_name', med.file\_name\_original, 'original\_mime\_type', med.file\_mime\_type) ORDER BY ssm\_link.display\_order) FROM public.shop\_service\_media ssm\_link JOIN public.media med ON med.id = ssm\_link.media\_id LEFT JOIN ss\_media\_translations ssm\_tr ON ssm\_tr.media\_link\_id\_text = ssm\_link.id::TEXT WHERE ssm\_link.shop\_service\_waypoint\_id = sssd.waypoint\_id) AS media\_gallery,

sssd.data\_last\_verified\_at,

sssd.deleted\_at,

sssd.created\_at,

creator\_profile.full\_name AS created\_by\_full\_name,

sssd.updated\_at,

updater\_profile.full\_name AS updated\_by\_full\_name,

wp.name AS parent\_waypoint\_name, -- English name from waypoints

wp.slug AS parent\_waypoint\_slug

FROM

public.shops\_and\_services\_details sssd

JOIN

public.waypoints wp ON sssd.waypoint\_id = wp.id

JOIN

public.shop\_service\_types\_master sstm ON sssd.service\_type\_id = sstm.id

LEFT JOIN

public.establishment\_price\_ranges\_master eprm ON sssd.general\_price\_range\_id = eprm.id

LEFT JOIN

direct\_translations sssd\_tr ON sssd\_tr.waypoint\_id\_text = sssd.waypoint\_id::TEXT

LEFT JOIN

service\_type\_translations st\_tr ON st\_tr.master\_id\_text = sstm.id::TEXT

LEFT JOIN

price\_range\_translations epr\_tr ON epr\_tr.master\_id\_text = eprm.id::TEXT

LEFT JOIN

public.profiles creator\_profile ON sssd.created\_by\_profile\_id = creator\_profile.id

LEFT JOIN

public.profiles updater\_profile ON sssd.updated\_by\_profile\_id = updater\_profile.id;

COMMENT ON VIEW public.v\_waypoint\_shop\_service\_details\_localized IS 'Provides a denormalized and localized view of shop and service details, including types, operational info, media, and contact details. Version 1.0';

```

\* \* \* \* \*

5\. Key Dependencies

- `public.shops\_and\_services\_details` (V1.3)

- `public.waypoints` (V1.3+ assumed)

- `public.shop\_service\_types\_master` (V1.2)

- `public.establishment\_price\_ranges\_master` (V1.1)

- `public.languages\_master` (V2.1)

- `public.payment\_methods\_master` (V1.1)

- `public.media` (V2.2)

- `public.shop\_service\_media` (V1.0)

- `public.media\_roles\_master`

- `public.translations` (V2.1)

- `public.profiles`

- `public.content\_statuses\_master`

\* \* \* \* \*

6\. Performance Considerations

- Highly Complex: Similar to other detail views, this view involves many joins and aggregations, especially for translations and expanding array FKs (`languages\_spoken\_codes`, `payment\_method\_ids`).

- Translation and Array Expansion CTEs: These contribute significantly to complexity. Efficient indexing on `public.translations` and all master tables (PKs, `is\_active` flags) is paramount.

- GIN Indexes: GIN indexes on array columns in `shops\_and\_services\_details` help if filtering by those tags directly on the base table, but here we are expanding them for display.

- Materialized View Potential: Given the complexity, if API read performance for this data becomes an issue and data can tolerate some latency, converting this to a MATERIALIZED VIEW (refreshed periodically) would be a strong consideration.

- Filtering in View Query: The primary filter `WHERE waypoint\_id = {input\_id}` on the main `shops\_and\_services\_details` alias `sssd` will be efficient due to the PK.

\* \* \* \* \*

7\. RLS & Security Notes

- The view operates with `SECURITY INVOKER` (default) permissions.

- Data visibility is controlled by RLS policies on the underlying `public.shops\_and\_services\_details` and `public.waypoints` tables. The RLS on `shops\_and\_services\_details` includes a check for its own `deleted\_at IS NULL` field, in addition to the parent waypoint's publication status and `deleted\_at` status.

- Users will only see shop/service details they are permitted to access based on these combined RLS policies.

\* \* \* \* \*

8\. API Endpoints Supported

- Primarily supports: `/waypoints/{waypoint\_id}/shop\_service\_details` (GET).

- The API backend would query: `SELECT \* FROM public.v\_waypoint\_shop\_service\_details\_localized WHERE waypoint\_id = {input\_id};`.

- The `lang` API query parameter would be used by the backend to select the appropriate language from the `all\_translations` JSONB objects or use the direct primary language fields from the view.

\* \* \* \* \*

9\. Rationale for Creation

- Decoupling & Simplification: Abstracts the complex multi-table join, array unnesting, and localization logic from the API backend, providing a clean and denormalized data source.

- Consistency: Ensures a uniform data structure for shop and service details for API consumers.

- Maintainability: Centralizes data shaping logic within the database, making application code simpler and database schema changes easier to manage with respect to API contracts (by updating the view definition).