<https://gemini.google.com/u/1/app/e2231210a6330ec4>

<https://gemini.google.com/u/1/app/a3fdff8118a11194>

Here is the full updated production-ready specification for `public.characteristic\_tags\_master`.

### Updated Production-Ready Specification: `public.characteristic\_tags\_master`

1\. Purpose & Primary Use-Cases

This table serves as a master lookup for categorizing region characteristics (e.g., "mountainous," "coastal," "historic towns"). It provides unique, stable codes for these characteristics and allows association with non-translatable metadata like icons or display order. All display names and descriptions for these characteristic tags are managed in the `public.translations` table, ensuring consistent and translatable classification. This table is referenced by `public.regions.characteristics\_tags`.

Key User-Story Touchpoints:

- Pilgrim: Sees translated characteristic tags (names fetched from `translations`) on a region's page or as filter options.

- Admin/Manager: Uses characteristic codes to tag regions and manages primary language names/descriptions in `translations`. Manages codes, icons, and display order in this master table.

- Platform: Uses codes for filtering regions.

2\. Schema

| Column Name | Data Type | Constraints | Description |

| `code` | `TEXT` | `PRIMARY KEY` | Unique, stable code for the tag (e.g., 'mountainous'). Non-changing. Used as `row\_foreign\_key` in `translations`. |

| `icon\_identifier` | `TEXT` | | Optional identifier for a UI icon (e.g., 'icon-mountain'). |

| `display\_order` | `INTEGER` | `NOT NULL DEFAULT 0`, `UNIQUE` | For ordering tags in UI. Ensures deterministic sorting. |

| `is\_active` | `BOOLEAN` | `NOT NULL DEFAULT true` | Whether the tag is active and usable. |

| `created\_at` | `TIMESTAMPTZ` | `NOT NULL DEFAULT now()` | Timestamp of creation. |

| `updated\_at` | `TIMESTAMPTZ` | `NOT NULL DEFAULT now()` | Timestamp of last update. Auto-updated by trigger. |

| `created\_by\_profile\_id` | `UUID` | `REFERENCES public.profiles(id) ON DELETE SET NULL` | Profile ID of the user who created the record. |

| `updated\_by\_profile\_id` | `UUID` | `REFERENCES public.profiles(id) ON DELETE SET NULL` | Profile ID of the user who last updated the record. |

Translatable Fields (Managed in public.translations table):

The following conceptual fields for a characteristic tag will have their content stored in public.translations with table\_identifier = 'characteristic\_tags\_master' and row\_foreign\_key = characteristic\_tags\_master.code:

- `name` (e.g., `column\_identifier = 'name'`)

- `description` (e.g., `column\_identifier = 'description'`, optional)

3\. PostgreSQL DDL

SQL

```

CREATE TABLE IF NOT EXISTS public.characteristic\_tags\_master (

code TEXT PRIMARY KEY,

icon\_identifier TEXT,

display\_order INTEGER NOT NULL DEFAULT 0 UNIQUE,

is\_active BOOLEAN NOT NULL DEFAULT true,

created\_at TIMESTAMPTZ NOT NULL DEFAULT now(),

updated\_at TIMESTAMPTZ NOT NULL DEFAULT now(),

created\_by\_profile\_id UUID REFERENCES public.profiles(id) ON DELETE SET NULL,

updated\_by\_profile\_id UUID REFERENCES public.profiles(id) ON DELETE SET NULL

);

COMMENT ON TABLE public.characteristic\_tags\_master IS 'Master lookup for region characteristic tags. Names/descriptions are in the public.translations table. Referenced by regions.characteristics\_tags. Version: 2.0';

COMMENT ON COLUMN public.characteristic\_tags\_master.code IS 'Primary Key. Unique, stable code for the characteristic tag (e.g., ''mountainous'', ''historic\_towns''). Non-changing. Used as row\_foreign\_key in translations.';

COMMENT ON COLUMN public.characteristic\_tags\_master.icon\_identifier IS 'Optional identifier for a UI icon associated with this tag (e.g., ''icon-mountain'', ''icon-castle'').';

COMMENT ON COLUMN public.characteristic\_tags\_master.display\_order IS 'For ordering tags in UI dropdowns, filter lists, or displays. Ensures deterministic sorting.';

COMMENT ON COLUMN public.characteristic\_tags\_master.is\_active IS 'Indicates if the characteristic tag is active and available for use in the system.';

COMMENT ON COLUMN public.characteristic\_tags\_master.created\_at IS 'Timestamp of when the characteristic tag record was created.';

COMMENT ON COLUMN public.characteristic\_tags\_master.updated\_at IS 'Timestamp of when the characteristic tag record was last updated. Auto-updated by a trigger.';

COMMENT ON COLUMN public.characteristic\_tags\_master.created\_by\_profile\_id IS 'Profile ID of the user who created the characteristic tag record.';

COMMENT ON COLUMN public.characteristic\_tags\_master.updated\_by\_profile\_id IS 'Profile ID of the user who last updated the characteristic tag record.';

-- Indexing:

-- PRIMARY KEY on 'code' already creates a unique index. [cite: 535]

-- UNIQUE constraint on 'display\_order' already creates an index. [cite: 535]

-- Trigger for updated\_at (ensure public.handle\_updated\_at() or equivalent function exists)

CREATE TRIGGER trigger\_characteristic\_tags\_master\_set\_updated\_at

BEFORE UPDATE ON public.characteristic\_tags\_master

FOR EACH ROW

EXECUTE FUNCTION public.handle\_updated\_at(); -- Or extensions.moddatetime, or public.set\_current\_timestamp\_updated\_at [cite: 536]

-- Trigger for deleting related translations

-- (ensure public.cleanup\_related\_translations() function is adaptable for TEXT PKs

-- or a specific version is made that correctly uses OLD.code as the row\_foreign\_key

-- for table\_identifier = 'characteristic\_tags\_master')

CREATE TRIGGER trigger\_delete\_characteristic\_tag\_translations

AFTER DELETE ON public.characteristic\_tags\_master

FOR EACH ROW EXECUTE FUNCTION public.cleanup\_related\_translations(); [cite: 537]

-- RLS Policies

ALTER TABLE public.characteristic\_tags\_master ENABLE ROW LEVEL SECURITY; [cite: 538]

CREATE POLICY "Allow admin full access on characteristic\_tags\_master"

ON public.characteristic\_tags\_master

FOR ALL

USING (public.is\_platform\_admin()) -- Assumes helper function [cite: 539]

WITH CHECK (public.is\_platform\_admin());

CREATE POLICY "Allow authenticated users read access on characteristic\_tags\_master"

ON public.characteristic\_tags\_master

FOR SELECT

TO authenticated

USING (is\_active = true); [cite: 540]

CREATE POLICY "Allow anonymous users read access on characteristic\_tags\_master"

ON public.characteristic\_tags\_master

FOR SELECT

TO anon

USING (is\_active = true); [cite: 541]

```

4\. JSON Schema Mirror

JSON

```

{

"title": "characteristic\_tag\_master",

"description": "Master lookup for region characteristic tags. Names/descriptions are in 'translations' table. Version: 2.0",

"type": "object",

"properties": {

"code": {

"type": "string",

"description": "Primary Key. Unique, stable code for the tag (e.g., 'mountainous'). Used as 'row\_foreign\_key' in 'translations'.",

"pattern": "^[a-z0-9\_]+$"

},

"icon\_identifier": {

"type": ["string", "null"],

"description": "Optional identifier for a UI icon.",

"maxLength": 50 [cite: 543]

},

"display\_order": {

"type": "integer",

"default": 0,

"description": "Order for displaying tags in UI. Must be unique." [cite: 544]

},

"is\_active": {

"type": "boolean",

"default": true,

"description": "Indicates if the characteristic tag is active and available for use."

},

"created\_at": {

"type": "string",

"format": "date-time",

"readOnly": true

},

"updated\_at": {

"type": "string",

"format": "date-time",

"readOnly": true

},

"created\_by\_profile\_id": {

"type": ["string", "null"],

"format": "uuid",

"description": "Profile ID of the user who created the record."

},

"updated\_by\_profile\_id": {

"type": ["string", "null"],

"format": "uuid",

"description": "Profile ID of the user who last updated the record."

}

},

"required": [

"code",

"display\_order",

"is\_active"

]

}

```

5\. Relationships & Integrity

- Primary Key: `code` (TEXT).

- Referenced By:

- `public.regions.characteristics\_tags` (this column in regions will need to be changed from `region\_characteristic\_tag\_enum[]` to `TEXT[]`, and its values will be codes from this table).

- `public.translations.row\_foreign\_key` (conceptually, when `table\_identifier = 'characteristic\_tags\_master'`). This is managed by the `cleanup\_related\_translations` trigger on delete and application logic for inserts/updates.

- Unique Constraints: `code` (PK), `display\_order`.

6\. Impact on regions Table

The regions table schema needs the characteristics\_tags column definition to be changed from region\_characteristic\_tag\_enum[] to TEXT[]. Data in regions.characteristics\_tags will need migration from old ENUM values to the new TEXT codes. The region\_characteristic\_tag\_enum type should be dropped after migration.

7\. Multilingual Strategy

All display names and descriptions are stored in public.translations. The characteristic\_tags\_master table itself contains no translatable text. To fetch a translated name for a tag code:

SQL

```

SELECT translated\_text

FROM public.translations

WHERE table\_identifier = 'characteristic\_tags\_master'

AND column\_identifier = 'name' -- Or 'description'

AND row\_foreign\_key = 'tag\_code\_value' -- e.g., 'mountainous'

AND language\_code = 'user\_language\_code'

AND translation\_status = 'published\_live'; -- Or relevant status

```

8\. Role-Based Workflow & RLS Notes

- Content Management: Core records (`code`, `icon\_identifier`, `display\_order`, `is\_active`) are managed by Administrators. Textual names/descriptions are managed via `public.translations` workflow.

- RLS Policies:

- Admins: Full access via `public.is\_platform\_admin()`.

- Authenticated/Anonymous Users: Read access to `is\_active = true` tags. Access to translated text is governed by RLS on `public.translations`.

9\. ENUM vs Lookup Discussion

This table correctly implements the lookup table pattern, replacing the previous region\_characteristic\_tag\_enum. This allows richer attributes (icons, order) and proper i18n of names/descriptions via public.translations.

10\. UI/UX Enablement

- UI uses `code` for internal logic and fetching translations.

- `icon\_identifier` and `display\_order` drive visual presentation.

- Translated `name` (and `description`) from `public.translations` are displayed to users.

- Enables filtering regions by characteristics.

11\. Scalability & Future-Proofing

The structure is scalable for languages and flexible for adding future non-translatable attributes. The table size is expected to remain small.

12\. Next-Action Checklist

- 🔴 Define Helper Function: Ensure `public.is\_platform\_admin()` is defined and securely implemented.

- 🔴 Define `updated\_at` Trigger Function: Ensure a standard `public.handle\_updated\_at()` (or chosen equivalent like `extensions.moddatetime`) is defined and used consistently.

- 🔴 Verify/Adapt `cleanup\_related\_translations`: Confirm the generic `public.cleanup\_related\_translations()` function correctly handles `TEXT` primary keys (`OLD.code`) for the `row\_foreign\_key` when `TG\_TABLE\_NAME` is `characteristic\_tags\_master`. Document its exact parameters.

- 🟠 Deploy DDL: Apply the `CREATE TABLE` and `COMMENT` DDL for `public.characteristic\_tags\_master`.

- 🟠 Populate Initial Data: Insert seed data for `code`, `icon\_identifier`, `display\_order`, `is\_active` (e.g., {'mountainous', 'icon-peaks', 10, true}).

- 🟠 Populate Translations: For each seed `code`, create corresponding `name` (and `description`, if used) entries in `public.translations` for primary and supported languages.

- 🟠 Implement Triggers: Create `trigger\_characteristic\_tags\_master\_set\_updated\_at` and `trigger\_delete\_characteristic\_tag\_translations`.

- 🟠 Implement RLS Policies: Apply the RLS policies as defined.

- 🟢 Update `regions` Table: Plan modification of `regions.characteristics\_tags` to `TEXT[]` and data migration.

- 🟢 Drop Old ENUM: Plan `DROP TYPE IF EXISTS public.region\_characteristic\_tag\_enum` after `regions` table migration.