## [**https://gemini.google.com/u/1/app/980a7af1dc7f079a**](https://gemini.google.com/u/1/app/980a7af1dc7f079a)

<https://gemini.google.com/u/1/app/117b6ecbb3fe7734>

\* \* \* \* \*

Updated Production-Ready Specification: Tags Master Table (Version 1.3)

-----------------------------------------------------------------------

This document details the structure, purpose, and considerations for the `tags\_master` table. This table provides a controlled vocabulary for granularly tagging various entities, primarily waypoints and events. Version 1.3 updates RLS policies to align with the platform-wide security architecture. Version 1.2 incorporated the `is\_active` flag and standard V2 audit/indexing enhancements.

1\. Purpose & Primary Use-Cases

The `tags\_master` table stores a curated list of descriptive tags. For waypoints, these are used via `waypoints.waypoint\_subcategory\_tag\_ids` (e.g., "Franciscan Site"). For events, tags can define themes via `events\_details.event\_theme\_or\_focus\_tag\_ids` (e.g., "Medieval Festival"). It aims to standardize sub-categorization, support multilingual display, enable faceted search, and improve data consistency.

2\. Updated Schema (Markdown Table)

| column | data\_type | constraints | description |

| id | integer | Primary Key (Generated as identity always) | Unique identifier for the tag. |

| tag\_code | text | Unique, Not Null, CHECK (length(tag\_code) > 0 AND length(tag\_code) &lt;= 50 AND tag\_code ~ '^[a-z0-9\_]+$') | Short, stable, machine-readable code for the tag (e.g., 'franciscan\_site'). Snake\_case, max 50 chars. |

| label | text | Not Null, CHECK (length(label) > 0 AND length(label) &lt;= 100) | Human-readable name for the tag. (Translatable via `public.translations`). Primary reference language (English). Max 100 chars. |

| description | text | Nullable | Optional description of the tag. (Translatable via `public.translations`). Primary reference language (English). |

| tag\_type | text | Nullable, CHECK (tag\_type IS NULL OR (length(tag\_type) > 0 AND length(tag\_type) &lt;= 50 AND tag\_type ~ '^[a-z0-9\_]+$')) | Optional: A way to group tags (e.g., 'amenity', 'event\_theme'). Snake\_case, max 50 chars. |

| icon\_identifier | text | Nullable, CHECK (icon\_identifier IS NULL OR length(icon\_identifier) &lt;= 100) | Optional: Name, class, or path for a UI icon associated with this tag. Max 100 chars. |

| sort\_order | integer | Not Null, Default 0 | Determines the display order of tags in UI lists or filters. Lower numbers appear first. |

| is\_active | boolean | Not Null, Default true | True if the tag is active and available for use; false if retired. |

| created\_at | timestamp with time zone | Not Null, Default now() | Timestamp of record creation. |

| updated\_at | timestamp with time zone | Not Null, Default now() | Timestamp of last update (auto-updated by trigger). |

| created\_by\_profile\_id | uuid | Nullable, Foreign Key to profiles(id) ON DELETE SET NULL | Profile ID of the user who created the record. |

| updated\_by\_profile\_id | uuid | Nullable, Foreign Key to profiles(id) ON DELETE SET NULL | Profile ID of the user who last updated the record. |

3\. PostgreSQL DDL

SQL

```

-- Assumes public.profiles table exists and has a `roles TEXT[]` column.

-- Assumes RLS helper function public.has\_role(TEXT) RETURNS BOOLEAN exists.

-- Assumes public.set\_current\_timestamp\_updated\_at() function exists.

-- Assumes public.cleanup\_related\_translations(TEXT) function exists.

CREATE TABLE public.tags\_master (

id SERIAL PRIMARY KEY,

tag\_code TEXT UNIQUE NOT NULL CHECK (length(tag\_code) > 0 AND length(tag\_code) <= 50 AND tag\_code ~ '^[a-z0-9\_]+$'),

label TEXT NOT NULL CHECK (length(label) > 0 AND length(label) <= 100),

description TEXT NULL,

tag\_type TEXT NULL CHECK (tag\_type IS NULL OR (length(tag\_type) > 0 AND length(tag\_type) <= 50 AND tag\_type ~ '^[a-z0-9\_]+$')),

icon\_identifier TEXT NULL CHECK (icon\_identifier IS NULL OR length(icon\_identifier) <= 100),

sort\_order INTEGER NOT NULL DEFAULT 0,

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 NULL REFERENCES public.profiles(id) ON DELETE SET NULL,

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

);

COMMENT ON TABLE public.tags\_master IS 'Master list of descriptive tags for various entities (e.g., waypoints, events). `label` and `description` are translatable. Version 1.3';

COMMENT ON COLUMN public.tags\_master.id IS 'Unique identifier for the tag.';

COMMENT ON COLUMN public.tags\_master.tag\_code IS 'Short, stable, machine-readable code for the tag (snake\_case). Max 50 chars. E.g., ''franciscan\_site'', ''medieval\_theme''.';

COMMENT ON COLUMN public.tags\_master.label IS 'Human-readable English label for UI and translation base. Max 100 chars. (Translatable via public.translations).';

COMMENT ON COLUMN public.tags\_master.description IS 'Optional English description of the tag. (Translatable via public.translations).';

COMMENT ON COLUMN public.tags\_master.tag\_type IS 'Optional: Type to group tags (e.g., ''amenity'', ''poi\_feature'', ''event\_theme''). Snake\_case, max 50 chars. The label for the tag\_type itself can be translated if needed by referencing its code in the translations table.';

COMMENT ON COLUMN public.tags\_master.icon\_identifier IS 'Optional: Name, class, or path for a UI icon associated with this tag. Max 100 chars.';

COMMENT ON COLUMN public.tags\_master.sort\_order IS 'Determines display order in UI lists/filters, potentially within a tag\_type.';

COMMENT ON COLUMN public.tags\_master.is\_active IS 'True if the tag is active and available for use; false if retired. Defaults to true.';

COMMENT ON COLUMN public.tags\_master.created\_at IS 'Timestamp of record creation.';

COMMENT ON COLUMN public.tags\_master.updated\_at IS 'Timestamp of last update (auto-updated by trigger).';

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

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

-- Indexes

CREATE INDEX IF NOT EXISTS idx\_tags\_master\_created\_by ON public.tags\_master(created\_by\_profile\_id);

CREATE INDEX IF NOT EXISTS idx\_tags\_master\_updated\_by ON public.tags\_master(updated\_by\_profile\_id);

CREATE INDEX IF NOT EXISTS idx\_tags\_master\_tag\_type ON public.tags\_master(tag\_type) WHERE tag\_type IS NOT NULL;

CREATE INDEX IF NOT EXISTS idx\_tags\_master\_sort\_order ON public.tags\_master(sort\_order);

CREATE INDEX IF NOT EXISTS idx\_tags\_master\_is\_active ON public.tags\_master(is\_active);

CREATE INDEX IF NOT EXISTS idx\_tags\_master\_label ON public.tags\_master(label); -- For sorting by label

-- Trigger for updated\_at

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

BEFORE UPDATE ON public.tags\_master

FOR EACH ROW

EXECUTE FUNCTION public.set\_current\_timestamp\_updated\_at();

COMMENT ON TRIGGER trigger\_tags\_master\_set\_updated\_at ON public.tags\_master IS 'Trigger to automatically update updated\_at timestamp on row modification.';

-- Trigger for orphan translation cleanup

CREATE TRIGGER trigger\_cleanup\_tags\_master\_translations

AFTER DELETE ON public.tags\_master

FOR EACH ROW

EXECUTE FUNCTION public.cleanup\_related\_translations('tags\_master');

COMMENT ON TRIGGER trigger\_cleanup\_tags\_master\_translations ON public.tags\_master IS 'Cleans up orphaned translations when a tag is deleted. The trigger function should pass OLD.id::TEXT as the row foreign key.';

```

4\. JSON Schema Mirror

JSON

```

{

"title": "tag\_master",

"description": "Master list of descriptive tags for various entities. `label` and `description` are translatable. Version 1.3",

"type": "object",

"properties": {

"id": {

"type": "integer",

"description": "Unique identifier for the tag. Primary Key.",

"readOnly": true

},

"tag\_code": {

"type": "string",

"description": "Short, stable, machine-readable code for the tag (snake\_case). Max 50 chars.",

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

"maxLength": 50

},

"label": {

"type": "string",

"description": "Human-readable English label for UI and translation base. Max 100 chars. (Translatable via public.translations).",

"maxLength": 100

},

"description": {

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

"description": "Optional English description of the tag. (Translatable via public.translations)."

},

"tag\_type": {

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

"description": "Optional: Type to group tags (e.g., 'amenity', 'poi\_feature', 'event\_theme'). Snake\_case, max 50 chars.",

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

"maxLength": 50

},

"icon\_identifier": {

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

"maxLength": 100,

"description": "Optional: Name, class, or path for a UI icon associated with this tag."

},

"sort\_order": {

"type": "integer",

"default": 0,

"description": "Determines display order in UI lists/filters."

},

"is\_active": {

"type": "boolean",

"default": true,

"description": "True if the tag is active and available for use; false if retired."

},

"created\_at": {

"type": "string",

"format": "date-time",

"description": "Timestamp of record creation.",

"readOnly": true

},

"updated\_at": {

"type": "string",

"format": "date-time",

"description": "Timestamp of last update.",

"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": [

"tag\_code",

"label",

"sort\_order",

"is\_active"

]

}

```

5\. Relationships & Integrity

- Primary Key: `id` (SERIAL -> INTEGER)

- Unique Constraint: `tag\_code` must be unique.

- Foreign Key References FROM other tables:

- `waypoints.waypoint\_subcategory\_tag\_ids` (an `INTEGER[]` column) will contain IDs that reference `tags\_master.id`.

- `events\_details.event\_theme\_or\_focus\_tag\_ids` (an `INTEGER[]` column) will contain IDs that reference `tags\_master.id`.

- 🔴 Integrity Note: PostgreSQL does not natively enforce referential integrity for individual elements within an `INTEGER[]` array. Database triggers \*must\* be implemented on referencing tables (e.g., `waypoints`, `events\_details`) to ensure all IDs in their tag arrays correspond to valid, `is\_active = true` `id` entries in `tags\_master`.

- Foreign Key References TO other tables:

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

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

6\. Multilingual Strategy

- Translatable Fields:

- `label`: The English name for the tag. (Translatable via `public.translations`)

- `description`: The English description for the tag. (Translatable via `public.translations`)

- Non-Translatable Fields:

- `tag\_code` is a stable identifier and is not translated.

- `tag\_type` codes are stable; their user-facing labels can be translated by adding entries to `public.translations` keyed by the `tag\_type` code if needed.

- Translation Management:

- An `AFTER DELETE` trigger (`trigger\_cleanup\_tags\_master\_translations`) is implemented to call `public.cleanup\_related\_translations` to remove orphaned translations.

7\. Role-Based Workflow & RLS Notes

- Content Management: `tags\_master` provides a controlled vocabulary. Users with `platform\_admin` or `admin` roles typically manage the definitions of these tags. Other roles like `regional\_content\_manager` or `accommodation\_host` would \*use\* these tags when creating or updating content they manage.

- Lifecycle: Tags are made inactive for new assignments by setting `is\_active = false`. This is the preferred method over deletion, especially if a tag has been used, to maintain historical data integrity.

- RLS Policies:

- Enable RLS:

SQL

```

ALTER TABLE public.tags\_master ENABLE ROW LEVEL SECURITY;

```

- Anonymous & Authenticated Users (Read-Only for Active Tags):

SQL

```

CREATE POLICY "Allow public read access to active tags"

ON public.tags\_master FOR SELECT

USING (is\_active = true);

```

- Platform Administrators & Administrators (Full Control):

SQL

```

CREATE POLICY "Allow platform\_admins and admins full access to tags"

ON public.tags\_master FOR ALL

USING (public.has\_role('platform\_admin') OR public.has\_role('admin'))

WITH CHECK (public.has\_role('platform\_admin') OR public.has\_role('admin'));

```

- Content Managers (Read-Only for All Tags - Example): Content managers need to see all available tags (even inactive ones, perhaps, to understand historical data or if they have a role in suggesting new tags). For assigning tags, they would typically only see active ones via UI filtering.

SQL

```

CREATE POLICY "Allow content\_roles read access to all tags"

ON public.tags\_master FOR SELECT

USING (public.has\_role('regional\_content\_manager') OR public.has\_role('admin') OR public.has\_role('platform\_admin')); -- Adjust roles as needed

```

- Note: Ensure the `public.has\_role(TEXT)` function correctly checks the `roles` array in the `public.profiles` table for the current `auth.uid()`. The RLS strategy implies that only `platform\_admin` or `admin` can create/update/delete these master tags.

8\. ENUM vs Lookup Discussion

- 🟢 Decision: This `tags\_master` table correctly promotes a free-text array or simpler ENUM array concept for sub-categorization and theming.

- Reasoning: A master lookup table provides standardization, richness (attributes like `tag\_type`, `icon\_identifier`, `is\_active`), i18n, maintainability, data integrity, and auditability.

9\. UI/UX Enablement

- `label` (translated): Used for display names in filter checkboxes, tag clouds, and on detail pages.

- `icon\_identifier` (optional): Can be used to display small icons next to tags.

- `tag\_type`: Can be used by the UI to group related tags in filter sections (e.g., "Filter by Amenity," "Filter by Event Theme").

- `sort\_order`: Ensures a logical and consistent presentation of tags.

- `is\_active`: UI should primarily display/allow selection of active tags for new/updated entities.

10\. Key Considerations & Definitions

- Granularity & Scope: `tag\_type` helps scope tags (e.g., some for POIs, some for events).

- `tag\_type` Usage: Clearly define `tag\_type` values. 'event\_theme' explicitly supports event tagging.

- Deletion Strategy: Retiring tags by setting `is\_active = false` is the primary strategy. Physical deletion should be rare and managed by an admin, considering existing references. The array FK validation triggers on referencing tables must check `tags\_master.is\_active = true`.

11\. Scalability & Future-Proofing

- Lookup Table Structure: Scales well. Performance for referencing tables depends on GIN indexes on their array FK columns.

- Audit Fields & `is\_active` flag: Provide change history and robust lifecycle management.

12\. Next-Action Checklist

- 🔴 Define Global RLS Helper Functions: Ensure `public.has\_role(TEXT)` is implemented securely and available.

- 🔴 Create Table: Execute the DDL for `public.tags\_master` (Version 1.3).

- 🔴 Apply Triggers: Ensure `set\_current\_timestamp\_updated\_at` and `cleanup\_related\_translations` triggers are created and applied.

- 🔴 Database Triggers for Referencing Tables: Critically ensure array FK validation triggers are implemented on `waypoints` (for `waypoint\_subcategory\_tag\_ids`) and `events\_details` (for `event\_theme\_or\_focus\_tag\_ids`) to validate against `tags\_master.id` AND `tags\_master.is\_active = true`.

- 🔴 Initial Population: Insert an initial set of agreed-upon tags, including examples for different `tag\_type` values. Ensure `is\_active = true` and `created\_by\_profile\_id` are set.

- 🟠 RLS Policies: Implement and thoroughly test the RLS policies defined in section 7.

- 🟢 Iconography & UI Grouping: Coordinate with UI team based on `icon\_identifier` and `tag\_type`.

- 🟢 Translation Entries: Prepare initial English entries for `label` and `description` in `public.translations`.

\* \* \* \* \*