<https://gemini.google.com/u/1/app/bf28f389cd093e55?is_sa=1&android-min-version=301356232&ios-min-version=322.0&campaign_id=bkws&utm_source=google&utm_medium=cpc&utm_campaign=2024enUS_gemfeb&pt=9008&mt=8&ct=p-growth-sem-bkws>

<https://gemini.google.com/u/1/app/1d4834db0819faf2>

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

### 3\. Updated Production-Ready Specification (`food\_water\_sources\_details`)

This document details the structure, purpose, and considerations for the `food\_water\_sources\_details` table, Version 1.3.1 (reflecting RLS role name alignment). This version incorporates full V2 enhancements including robust audit fields, refined i18n strategy, array foreign key integrity with active status checks, and media linking. It does \*not\* include denormalized English labels from master tables.

1\. Purpose & Primary Use-Cases

The `food\_water\_sources\_details` table stores specific attributes for waypoints that are food establishments (restaurants, cafes, grocery stores) or water sources (fountains, springs). It extends the generic `waypoints` entry to include crucial details such as the type of source, potability and reliability of water, price range for commercial establishments, types of meals served, dietary options, payment methods, and opening hours. This information is vital for pilgrims for logistical planning, ensuring they can find sustenance and safe drinking water along their route.

Key user-story touchpoints:

- Pilgrim: Finding places to eat or resupply groceries along the trail.

- Pilgrim: Locating reliable sources of potable water.

- Pilgrim: Filtering food options by type (restaurant, cafe), price range, or dietary accommodations (vegetarian, gluten-free).

- Pilgrim: Checking opening hours for food establishments or availability notes for water sources.

- Content Manager/Admin: Adding and maintaining detailed, accurate information about food and water points.

2\. Schema (`food\_water\_sources\_details`)

| column | data\_type | constraints | description |

| `waypoint\_id` | `BIGINT` | Primary Key, Foreign Key to `public.waypoints(id)` ON DELETE CASCADE | Links to the generic `waypoints` table. This is the PK. |

| `source\_type\_id` | `INTEGER` | Not Null, Foreign Key to `public.food\_water\_source\_types\_master(id)` ON DELETE RESTRICT | Type of food or water source, linking to an \*active\* `food\_water\_source\_types\_master` record. |

| `is\_potable\_water\_source` | `BOOLEAN` | Not Null, Default `false` | Explicitly true if this is a source of drinkable water. Defaults to false (not potable) unless explicitly stated. Primarily for non-commercial types. |

| `water\_reliability\_id` | `INTEGER` | Nullable, Foreign Key to `public.water\_reliability\_types\_master(id)` ON DELETE SET NULL | Reliability if it's a natural/public water source, linking to an \*active\* `water\_reliability\_types\_master` record. |

| `water\_source\_access\_notes` | `TEXT` | Nullable | Specific notes on finding or accessing a public water source (e.g., "behind the church," "tap is on a timer"). Primary reference language (English) text. (Translatable via `public.translations`) |

| `establishment\_price\_range\_id` | `INTEGER` | Nullable, Foreign Key to `public.establishment\_price\_ranges\_master(id)` ON DELETE SET NULL | General price range if it's a commercial establishment, linking to an \*active\* `establishment\_price\_ranges\_master` record. |

| `serves\_meal\_type\_tag\_ids` | `INTEGER[]` | Nullable | Array of FKs to `meal\_type\_tags\_master(id)`. Types of meals served if commercial (e.g., breakfast, lunch, dinner). Integrity (existence and active status) enforced by DB trigger. |

| `highlighted\_dishes\_local\_specialties` | `TEXT[]` | Nullable | Array of 1-3 key dishes or local specialties offered (free text). Elements are primary reference language (English) text. (Translatable via `public.translations` for each element) |

| `dietary\_option\_tag\_ids` | `INTEGER[]` | Nullable | Array of FKs to `dietary\_option\_tags\_master(id)`. Available dietary options (e.g., vegetarian, gluten-free). Integrity (existence and active status) enforced by DB trigger. |

| `opening\_hours\_structured` | `JSONB` | Nullable, CHECK (`opening\_hours\_structured` IS NULL OR jsonb\_type\_of(`opening\_hours\_structured`) IN ('object', 'array')) | Structured opening hours for commercial establishments (e.g., based on schema.org OpeningHoursSpecification). Adheres to the centrally defined project schema. |

| `opening\_hours\_text\_notes` | `TEXT` | Nullable | Human-readable summary or additional notes about opening hours/availability (e.g., "Closed on Tuesdays," "Siesta 13:00-16:00"). Primary reference language (English) text. (Translatable via `public.translations`) |

| `opening\_hours\_last\_verified\_at` | `TIMESTAMPTZ` | Nullable | When the opening hours were last checked/verified by an editor or reliable source. |

| `outdoor\_seating\_available` | `BOOLEAN` | Nullable | Does the commercial establishment offer outdoor seating? Nullable allows for "unknown" or not applicable. |

| `payment\_method\_tag\_ids` | `INTEGER[]` | Nullable | Array of FKs to `payment\_methods\_master(id)`. Indicates accepted payment methods if commercial. Integrity (existence and active status) enforced by DB trigger. |

| `specific\_notes\_for\_pilgrims` | `TEXT` | Nullable | Any notes specifically for pilgrims (e.g., "Pilgrim menu available," "Offers discount with credential"). Primary reference language (English) text. (Translatable via `public.translations`) |

| `data\_last\_verified\_at` | `TIMESTAMPTZ` | Nullable | When the overall details for this source were last verified by an editor or reliable source. |

| `created\_at` | `TIMESTAMPTZ` | Not Null, Default `now()` | Timestamp of record creation. |

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

| `updated\_at` | `TIMESTAMPTZ` | Not Null, Default `now()` | Timestamp of last update (auto-updated by trigger). |

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

3\. PostgreSQL DDL

SQL

```

-- Ensure prerequisite tables are V2 compliant and created first:

-- public.waypoints (BIGINT id PK, content\_visibility\_status\_id INTEGER, deleted\_at TIMESTAMPTZ)

-- public.profiles (UUID id PK, roles TEXT[])

-- public.media (UUID id PK)

-- public.media\_roles\_master (TEXT code PK)

-- public.translations (for i18n)

-- public.languages\_master (for i18n)

-- public.content\_statuses\_master (INTEGER id PK, is\_publicly\_visible BOOLEAN)

-- Prerequisite Master Tables (ensure these match their full V1.1/V1.2 specs previously defined)

-- public.food\_water\_source\_types\_master (Version 1.2.1 - with label, is\_active, audit, triggers, RLS)

-- public.water\_reliability\_types\_master (Version 1.2.1 - with label, is\_active, audit, triggers, RLS)

-- public.establishment\_price\_ranges\_master (Version 1.1 - with label, is\_active, audit, triggers, RLS)

-- public.meal\_type\_tags\_master (Version 1.1 - with label, is\_active, audit, triggers, RLS)

-- public.dietary\_option\_tags\_master (Version 1.1 - with label, is\_active, audit, triggers, RLS)

-- public.payment\_methods\_master (Version 1.1 - with label, is\_active, audit, triggers, RLS)

-- (Full DDL for master tables should be referenced from their respective finalized specification documents)

-- Food & Water Sources Details Table (Version 1.3.1)

CREATE TABLE public.food\_water\_sources\_details (

waypoint\_id BIGINT PRIMARY KEY,

source\_type\_id INTEGER NOT NULL,

is\_potable\_water\_source BOOLEAN NOT NULL DEFAULT FALSE,

water\_reliability\_id INTEGER NULL,

water\_source\_access\_notes TEXT NULL,

establishment\_price\_range\_id INTEGER NULL,

serves\_meal\_type\_tag\_ids INTEGER[] NULL,

highlighted\_dishes\_local\_specialties TEXT[] NULL,

dietary\_option\_tag\_ids INTEGER[] NULL,

opening\_hours\_structured JSONB NULL CHECK (opening\_hours\_structured IS NULL OR jsonb\_type\_of(opening\_hours\_structured) IN ('object', 'array')),

opening\_hours\_text\_notes TEXT NULL,

opening\_hours\_last\_verified\_at TIMESTAMPTZ NULL,

outdoor\_seating\_available BOOLEAN NULL,

payment\_method\_tag\_ids INTEGER[] NULL,

specific\_notes\_for\_pilgrims TEXT NULL,

data\_last\_verified\_at TIMESTAMPTZ NULL,

created\_at TIMESTAMPTZ NOT NULL DEFAULT now(),

created\_by\_profile\_id UUID NULL,

updated\_at TIMESTAMPTZ NOT NULL DEFAULT now(),

updated\_by\_profile\_id UUID NULL,

CONSTRAINT fk\_waypoint FOREIGN KEY(waypoint\_id) REFERENCES public.waypoints(id) ON DELETE CASCADE,

CONSTRAINT fk\_source\_type FOREIGN KEY(source\_type\_id) REFERENCES public.food\_water\_source\_types\_master(id) ON DELETE RESTRICT,

CONSTRAINT fk\_water\_reliability FOREIGN KEY(water\_reliability\_id) REFERENCES public.water\_reliability\_types\_master(id) ON DELETE SET NULL,

CONSTRAINT fk\_establishment\_price\_range FOREIGN KEY(establishment\_price\_range\_id) REFERENCES public.establishment\_price\_ranges\_master(id) ON DELETE SET NULL,

CONSTRAINT fk\_created\_by\_profile FOREIGN KEY(created\_by\_profile\_id) REFERENCES public.profiles(id) ON DELETE SET NULL,

CONSTRAINT fk\_updated\_by\_profile FOREIGN KEY(updated\_by\_profile\_id) REFERENCES public.profiles(id) ON DELETE SET NULL

);

COMMENT ON TABLE public.food\_water\_sources\_details IS 'Specific details for food establishments and water sources, extending the waypoints table. Version 1.3.1'; [cite: 421]

COMMENT ON COLUMN public.food\_water\_sources\_details.waypoint\_id IS 'Links to the generic waypoints table. This is the PK.'; [cite: 422]

COMMENT ON COLUMN public.food\_water\_sources\_details.source\_type\_id IS 'FK to food\_water\_source\_types\_master. Type of food or water source. Referenced master record must be is\_active=true.';

COMMENT ON COLUMN public.food\_water\_sources\_details.is\_potable\_water\_source IS 'True if a source of drinkable water (especially for non-commercial types). Defaults to FALSE (not potable unless explicitly stated).'; [cite: 424, 425]

COMMENT ON COLUMN public.food\_water\_sources\_details.water\_reliability\_id IS 'FK to water\_reliability\_types\_master. Reliability if a natural/public water source. Referenced master record must be is\_active=true.';

COMMENT ON COLUMN public.food\_water\_sources\_details.water\_source\_access\_notes IS 'Specific notes on finding or accessing a public water source. Primary reference language (English) text. (Translatable via public.translations)'; [cite: 427]

COMMENT ON COLUMN public.food\_water\_sources\_details.establishment\_price\_range\_id IS 'FK to establishment\_price\_ranges\_master. General price range if commercial. Referenced master record must be is\_active=true.';

COMMENT ON COLUMN public.food\_water\_sources\_details.serves\_meal\_type\_tag\_ids IS 'Array of FKs to meal\_type\_tags\_master. Types of meals served if commercial. Integrity (existence and active status) of array elements MUST be enforced by a dedicated database trigger.'; [cite: 429]

COMMENT ON COLUMN public.food\_water\_sources\_details.highlighted\_dishes\_local\_specialties IS 'Array of key dishes or local specialties offered (free text). Elements are primary reference language (English) text. (Translatable via public.translations for each element).'; [cite: 431]

COMMENT ON COLUMN public.food\_water\_sources\_details.dietary\_option\_tag\_ids IS 'Array of FKs to dietary\_option\_tags\_master. Available dietary options. Integrity (existence and active status) of array elements MUST be enforced by a dedicated database trigger.'; [cite: 433]

COMMENT ON COLUMN public.food\_water\_sources\_details.opening\_hours\_structured IS 'Structured opening hours for commercial establishments in JSONB format. Adheres to the centrally defined schema.'; [cite: 434]

COMMENT ON COLUMN public.food\_water\_sources\_details.opening\_hours\_text\_notes IS 'Human-readable summary or notes about opening hours/availability. Primary reference language (English) text. (Translatable via public.translations)'; [cite: 436]

COMMENT ON COLUMN public.food\_water\_sources\_details.opening\_hours\_last\_verified\_at IS 'When the opening hours were last checked/verified.'; [cite: 437]

COMMENT ON COLUMN public.food\_water\_sources\_details.outdoor\_seating\_available IS 'Does the commercial establishment offer outdoor seating? Nullable for "unknown".'; [cite: 438]

COMMENT ON COLUMN public.food\_water\_sources\_details.payment\_method\_tag\_ids IS 'Array of FKs to payment\_methods\_master. Accepted payment methods if commercial. Integrity (existence and active status) of array elements MUST be enforced by a dedicated database trigger.'; [cite: 439]

COMMENT ON COLUMN public.food\_water\_sources\_details.specific\_notes\_for\_pilgrims IS 'Any notes specifically for pilgrims. Primary reference language (English) text. (Translatable via public.translations)'; [cite: 440]

COMMENT ON COLUMN public.food\_water\_sources\_details.data\_last\_verified\_at IS 'When the overall details for this source were last verified by an editor or reliable source.'; [cite: 441]

COMMENT ON COLUMN public.food\_water\_sources\_details.created\_at IS 'Timestamp of record creation.';

COMMENT ON COLUMN public.food\_water\_sources\_details.created\_by\_profile\_id IS 'Profile ID of the user who created this record.'; [cite: 442]

COMMENT ON COLUMN public.food\_water\_sources\_details.updated\_at IS 'Timestamp of last update (auto-updated by trigger).';

COMMENT ON COLUMN public.food\_water\_sources\_details.updated\_by\_profile\_id IS 'Profile ID of the user who last updated this record.'; [cite: 443]

-- Linking table for Food/Water Source Media (public.food\_water\_source\_media - Version 1.0)

CREATE TABLE IF NOT EXISTS public.food\_water\_source\_media (

id BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

food\_water\_source\_waypoint\_id BIGINT NOT NULL REFERENCES public.food\_water\_sources\_details(waypoint\_id) ON DELETE CASCADE,

media\_id UUID NOT NULL REFERENCES public.media(id) ON DELETE RESTRICT,

media\_role\_code TEXT NOT NULL REFERENCES public.media\_roles\_master(code) ON DELETE RESTRICT,

display\_order SMALLINT NOT NULL DEFAULT 0,

caption\_override TEXT NULL, -- Primary reference language (English). Translatable.

alt\_text\_override TEXT NULL, -- Primary reference language (English). Translatable.

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,

CONSTRAINT uq\_food\_water\_source\_media\_role\_order UNIQUE (food\_water\_source\_waypoint\_id, media\_id, media\_role\_code)

);

COMMENT ON TABLE public.food\_water\_source\_media IS 'Links food/water sources to media items for galleries or specific semantic roles. Version 1.0';

CREATE TRIGGER trigger\_food\_water\_source\_media\_set\_updated\_at

BEFORE UPDATE ON public.food\_water\_source\_media FOR EACH ROW EXECUTE FUNCTION public.set\_current\_timestamp\_updated\_at();

CREATE OR REPLACE FUNCTION public.cleanup\_food\_water\_source\_media\_translations()

RETURNS TRIGGER AS $$ BEGIN DELETE FROM public.translations WHERE table\_identifier = 'food\_water\_source\_media' AND row\_foreign\_key = OLD.id::TEXT; RETURN OLD; END; $$ LANGUAGE plpgsql SECURITY DEFINER;

CREATE TRIGGER trigger\_cleanup\_food\_water\_source\_media\_translations

AFTER DELETE ON public.food\_water\_source\_media FOR EACH ROW EXECUTE FUNCTION public.cleanup\_food\_water\_source\_media\_translations();

COMMENT ON TRIGGER trigger\_cleanup\_food\_water\_source\_media\_translations ON public.food\_water\_source\_media IS 'Cleans up orphaned translations for food\_water\_source\_media translatable fields.';

-- Triggers & Functions for food\_water\_sources\_details

CREATE TRIGGER trigger\_food\_water\_sources\_details\_set\_updated\_at

BEFORE UPDATE ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.set\_current\_timestamp\_updated\_at();

COMMENT ON TRIGGER trigger\_food\_water\_sources\_details\_set\_updated\_at ON public.food\_water\_sources\_details IS 'Trigger to automatically update updated\_at timestamp on row modification.'; [cite: 445]

CREATE OR REPLACE FUNCTION public.cleanup\_food\_water\_sources\_details\_translations()

RETURNS TRIGGER AS $$ BEGIN DELETE FROM public.translations WHERE table\_identifier = 'food\_water\_sources\_details' AND row\_foreign\_key = OLD.waypoint\_id::TEXT; DELETE FROM public.translations WHERE table\_identifier = 'food\_water\_source\_media' AND row\_foreign\_key IN (SELECT id::TEXT FROM public.food\_water\_source\_media WHERE food\_water\_source\_waypoint\_id = OLD.waypoint\_id); RETURN OLD; END; $$ LANGUAGE plpgsql SECURITY DEFINER;

CREATE TRIGGER trigger\_cleanup\_food\_water\_sources\_details\_translations

AFTER DELETE ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.cleanup\_food\_water\_sources\_details\_translations();

COMMENT ON TRIGGER trigger\_cleanup\_food\_water\_sources\_details\_translations ON public.food\_water\_sources\_details IS 'Cleans up orphaned translations for food\_water\_sources\_details and its linked media.';

-- Active Check Triggers for Single FKs

CREATE OR REPLACE FUNCTION public.check\_fwsrc\_source\_type\_active() RETURNS TRIGGER AS $$ DECLARE is\_active BOOLEAN; BEGIN IF NEW.source\_type\_id IS NOT NULL THEN SELECT fstm.is\_active INTO is\_active FROM public.food\_water\_source\_types\_master fstm WHERE fstm.id = NEW.source\_type\_id; IF NOT is\_active THEN RAISE EXCEPTION 'Referenced food\_water\_source\_types\_master record (id: %) must be active.', NEW.source\_type\_id; END IF; END IF; RETURN NEW; END; $$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_fwsrc\_source\_type\_active BEFORE INSERT OR UPDATE OF source\_type\_id ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.check\_fwsrc\_source\_type\_active();

CREATE OR REPLACE FUNCTION public.check\_fwsrc\_water\_reliability\_active() RETURNS TRIGGER AS $$ DECLARE is\_active BOOLEAN; BEGIN IF NEW.water\_reliability\_id IS NOT NULL THEN SELECT wrtm.is\_active INTO is\_active FROM public.water\_reliability\_types\_master wrtm WHERE wrtm.id = NEW.water\_reliability\_id; IF NOT is\_active THEN RAISE EXCEPTION 'Referenced water\_reliability\_types\_master record (id: %) must be active.', NEW.water\_reliability\_id; END IF; END IF; RETURN NEW; END; $$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_fwsrc\_water\_reliability\_active BEFORE INSERT OR UPDATE OF water\_reliability\_id ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.check\_fwsrc\_water\_reliability\_active();

CREATE OR REPLACE FUNCTION public.check\_fwsrc\_price\_range\_active() RETURNS TRIGGER AS $$ DECLARE is\_active BOOLEAN; BEGIN IF NEW.establishment\_price\_range\_id IS NOT NULL THEN SELECT eprm.is\_active INTO is\_active FROM public.establishment\_price\_ranges\_master eprm WHERE eprm.id = NEW.establishment\_price\_range\_id; IF NOT is\_active THEN RAISE EXCEPTION 'Referenced establishment\_price\_ranges\_master record (id: %) must be active.', NEW.establishment\_price\_range\_id; END IF; END IF; RETURN NEW; END; $$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_fwsrc\_price\_range\_active BEFORE INSERT OR UPDATE OF establishment\_price\_range\_id ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.check\_fwsrc\_price\_range\_active();

-- Array FK Integrity Triggers (checking existence and is\_active)

CREATE OR REPLACE FUNCTION public.check\_fwsrc\_serves\_meal\_types() RETURNS TRIGGER AS $$ DECLARE tag\_id INTEGER; is\_valid BOOLEAN; tag\_is\_active BOOLEAN; BEGIN IF NEW.serves\_meal\_type\_tag\_ids IS NOT NULL AND array\_length(NEW.serves\_meal\_type\_tag\_ids, 1) > 0 THEN FOREACH tag\_id IN ARRAY NEW.serves\_meal\_type\_tag\_ids LOOP SELECT EXISTS (SELECT 1 FROM public.meal\_type\_tags\_master WHERE id = tag\_id), mttm.is\_active INTO is\_valid, tag\_is\_active FROM public.meal\_type\_tags\_master mttm WHERE mttm.id = tag\_id; IF NOT is\_valid THEN RAISE EXCEPTION 'Invalid meal\_type\_tag\_id: %', tag\_id; END IF; IF NOT tag\_is\_active THEN RAISE EXCEPTION 'Inactive meal\_type\_tag\_id: %', tag\_id; END IF; END LOOP; END IF; RETURN NEW; END; $$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_fwsrc\_serves\_meal\_types BEFORE INSERT OR UPDATE OF serves\_meal\_type\_tag\_ids ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.check\_fwsrc\_serves\_meal\_types();

CREATE OR REPLACE FUNCTION public.check\_fwsrc\_dietary\_options() RETURNS TRIGGER AS $$ DECLARE tag\_id INTEGER; is\_valid BOOLEAN; tag\_is\_active BOOLEAN; BEGIN IF NEW.dietary\_option\_tag\_ids IS NOT NULL AND array\_length(NEW.dietary\_option\_tag\_ids, 1) > 0 THEN FOREACH tag\_id IN ARRAY NEW.dietary\_option\_tag\_ids LOOP SELECT EXISTS (SELECT 1 FROM public.dietary\_option\_tags\_master WHERE id = tag\_id), dotm.is\_active INTO is\_valid, tag\_is\_active FROM public.dietary\_option\_tags\_master dotm WHERE dotm.id = tag\_id; IF NOT is\_valid THEN RAISE EXCEPTION 'Invalid dietary\_option\_tag\_id: %', tag\_id; END IF; IF NOT tag\_is\_active THEN RAISE EXCEPTION 'Inactive dietary\_option\_tag\_id: %', tag\_id; END IF; END LOOP; END IF; RETURN NEW; END; $$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_fwsrc\_dietary\_options BEFORE INSERT OR UPDATE OF dietary\_option\_tag\_ids ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.check\_fwsrc\_dietary\_options();

CREATE OR REPLACE FUNCTION public.check\_fwsrc\_payment\_methods() RETURNS TRIGGER AS $$ DECLARE tag\_id INTEGER; is\_valid BOOLEAN; tag\_is\_active BOOLEAN; BEGIN IF NEW.payment\_method\_tag\_ids IS NOT NULL AND array\_length(NEW.payment\_method\_tag\_ids, 1) > 0 THEN FOREACH tag\_id IN ARRAY NEW.payment\_method\_tag\_ids LOOP SELECT EXISTS (SELECT 1 FROM public.payment\_methods\_master WHERE id = tag\_id), pmm.is\_active INTO is\_valid, tag\_is\_active FROM public.payment\_methods\_master pmm WHERE pmm.id = tag\_id; IF NOT is\_valid THEN RAISE EXCEPTION 'Invalid payment\_method\_tag\_id: %', tag\_id; END IF; IF NOT tag\_is\_active THEN RAISE EXCEPTION 'Inactive payment\_method\_tag\_id: %', tag\_id; END IF; END LOOP; END IF; RETURN NEW; END; $$ LANGUAGE plpgsql;

CREATE TRIGGER trigger\_check\_fwsrc\_payment\_methods BEFORE INSERT OR UPDATE OF payment\_method\_tag\_ids ON public.food\_water\_sources\_details FOR EACH ROW EXECUTE FUNCTION public.check\_fwsrc\_payment\_methods();

-- Indexes

CREATE INDEX idx\_fwsrc\_source\_type\_id ON public.food\_water\_sources\_details(source\_type\_id); [cite: 446]

CREATE INDEX idx\_fwsrc\_water\_reliability\_id ON public.food\_water\_sources\_details(water\_reliability\_id) WHERE water\_reliability\_id IS NOT NULL; [cite: 447]

CREATE INDEX idx\_fwsrc\_price\_range\_id ON public.food\_water\_sources\_details(establishment\_price\_range\_id) WHERE establishment\_price\_range\_id IS NOT NULL; [cite: 448]

CREATE INDEX idx\_fwsrc\_serves\_meal\_ids ON public.food\_water\_sources\_details USING GIN (serves\_meal\_type\_tag\_ids) WHERE serves\_meal\_type\_tag\_ids IS NOT NULL; [cite: 449]

CREATE INDEX idx\_fwsrc\_dietary\_option\_ids ON public.food\_water\_sources\_details USING GIN (dietary\_option\_tag\_ids) WHERE dietary\_option\_tag\_ids IS NOT NULL; [cite: 450]

CREATE INDEX idx\_fwsrc\_payment\_method\_ids ON public.food\_water\_sources\_details USING GIN (payment\_method\_tag\_ids) WHERE payment\_method\_tag\_ids IS NOT NULL; [cite: 451]

CREATE INDEX IF NOT EXISTS idx\_fwsrc\_opening\_hours\_structured ON public.food\_water\_sources\_details USING GIN (opening\_hours\_structured) WHERE opening\_hours\_structured IS NOT NULL;

CREATE INDEX idx\_fwsrc\_created\_by\_profile\_id ON public.food\_water\_sources\_details(created\_by\_profile\_id) WHERE created\_by\_profile\_id IS NOT NULL;

CREATE INDEX idx\_fwsrc\_updated\_by\_profile\_id ON public.food\_water\_sources\_details(updated\_by\_profile\_id) WHERE updated\_by\_profile\_id IS NOT NULL;

CREATE INDEX idx\_food\_water\_source\_media\_fw\_waypoint\_id ON public.food\_water\_source\_media(food\_water\_source\_waypoint\_id);

-- RLS Policies for food\_water\_sources\_details

ALTER TABLE public.food\_water\_sources\_details ENABLE ROW LEVEL SECURITY;

CREATE POLICY "public\_read\_food\_water\_sources\_details" ON public.food\_water\_sources\_details

FOR SELECT USING (

EXISTS (

SELECT 1 FROM public.waypoints w

JOIN public.content\_statuses\_master csm ON w.content\_visibility\_status\_id = csm.id

WHERE w.id = food\_water\_sources\_details.waypoint\_id AND csm.is\_publicly\_visible = TRUE AND w.deleted\_at IS NULL

)

);

CREATE POLICY "manage\_food\_water\_sources\_details\_for\_privileged\_users" ON public.food\_water\_sources\_details

FOR ALL USING (

((SELECT public.has\_role\_on\_profile(auth.uid(), 'platform\_admin')) OR (SELECT public.has\_role\_on\_profile(auth.uid(), 'regional\_content\_manager'))) -- Using consistent role names

) WITH CHECK (

((SELECT public.has\_role\_on\_profile(auth.uid(), 'platform\_admin')) OR (SELECT public.has\_role\_on\_profile(auth.uid(), 'regional\_content\_manager')))

AND (TG\_OP = 'INSERT' OR (TG\_OP = 'UPDATE' AND OLD.waypoint\_id = NEW.waypoint\_id OR (SELECT public.has\_role\_on\_profile(auth.uid(), 'platform\_admin'))))

);

-- (RLS for food\_water\_source\_media similar to other media linking tables, using consistent role names)

```

4\. JSON Schema Mirror (food\_water\_sources\_details)

(Reflects the schema table in Section 2; does not include denormalized \*\_label\_en columns)

JSON

```

{

"title": "food\_water\_source\_detail",

"description": "Specific details for food establishments and water sources, extending the waypoints table. Version 1.3.1",

"type": "object",

"properties": {

"waypoint\_id": {

"type": "integer",

"format": "int64",

"description": "Links to the generic waypoints table. PK."

},

"source\_type\_id": {

"type": "integer",

"description": "FK to food\_water\_source\_types\_master. Type of food or water source. Referenced master must be active."

},

"is\_potable\_water\_source": {

"type": "boolean",

"default": false,

"description": "True if this is a source of drinkable water."

},

"water\_reliability\_id": {

"type": ["integer", "null"],

"description": "FK to water\_reliability\_types\_master. Reliability if a natural/public water source. Referenced master must be active."

},

"water\_source\_access\_notes": {

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

"description": "Specific notes on accessing a public water source. Primary reference language (English) text. (Translatable via public.translations)"

},

"establishment\_price\_range\_id": {

"type": ["integer", "null"],

"description": "FK to establishment\_price\_ranges\_master. General price range if commercial. Referenced master must be active."

},

"serves\_meal\_type\_tag\_ids": {

"type": ["array", "null"],

"items": { "type": "integer" },

"description": "Array of FKs to meal\_type\_tags\_master. Types of meals served. Integrity & active status of tags enforced by DB trigger."

},

"highlighted\_dishes\_local\_specialties": {

"type": ["array", "null"],

"items": { "type": "string" },

"description": "Array of key dishes/specialties (free text). Elements are primary reference language (English) text. (Translatable via public.translations for each element)"

},

"dietary\_option\_tag\_ids": {

"type": ["array", "null"],

"items": { "type": "integer" },

"description": "Array of FKs to dietary\_option\_tags\_master. Dietary options. Integrity & active status of tags enforced by DB trigger."

},

"opening\_hours\_structured": {

"type": ["object", "array", "null"],

"description": "Structured opening hours (e.g., schema.org). Can be object or array of objects."

},

"opening\_hours\_text\_notes": {

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

"description": "Human-readable notes on opening hours. Primary reference language (English) text. (Translatable via public.translations)"

},

"opening\_hours\_last\_verified\_at": {

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

"format": "date-time",

"description": "When opening hours were last verified."

},

"outdoor\_seating\_available": {

"type": ["boolean", "null"],

"description": "Does the commercial establishment offer outdoor seating? Null for 'unknown'."

},

"payment\_method\_tag\_ids": {

"type": ["array", "null"],

"items": { "type": "integer" },

"description": "Array of FKs to payment\_methods\_master. Accepted payment methods. Integrity & active status of tags enforced by DB trigger."

},

"specific\_notes\_for\_pilgrims": {

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

"description": "Notes specifically for pilgrims. Primary reference language (English) text. (Translatable via public.translations)"

},

"data\_last\_verified\_at": {

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

"format": "date-time",

"description": "When overall details were last verified."

},

"created\_at": {

"type": "string",

"format": "date-time",

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

"readOnly": true

},

"created\_by\_profile\_id": {

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

"format": "uuid",

"description": "Profile ID of the creator."

},

"updated\_at": {

"type": "string",

"format": "date-time",

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

"readOnly": true

},

"updated\_by\_profile\_id": {

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

"format": "uuid",

"description": "Profile ID of the last updater."

}

},

"required": [

"waypoint\_id",

"source\_type\_id",

"is\_potable\_water\_source",

"created\_at",

"updated\_at"

]

}

```

5\. Relationships & Integrity

- Primary Key: `waypoint\_id` (`BIGINT`).

- Foreign Keys:

- `source\_type\_id` REFERENCES `public.food\_water\_source\_types\_master(id)` ON DELETE RESTRICT. (Integrity: Referenced master record must have `is\_active = true`, enforced by `trigger\_check\_fwsrc\_source\_type\_active`).

- `water\_reliability\_id` REFERENCES `public.water\_reliability\_types\_master(id)` ON DELETE SET NULL. (Integrity: Referenced master record must have `is\_active = true`, enforced by `trigger\_check\_fwsrc\_water\_reliability\_active`).

- `establishment\_price\_range\_id` REFERENCES `public.establishment\_price\_ranges\_master(id)` ON DELETE SET NULL. (Integrity: Referenced master record must have `is\_active = true`, enforced by `trigger\_check\_fwsrc\_price\_range\_active`).

- Audit FKs (`created\_by\_profile\_id`, `updated\_by\_profile\_id`) to `public.profiles(id)` ON DELETE SET NULL.

- Array Foreign Keys:

- `serves\_meal\_type\_tag\_ids` (elements `INTEGER`) to `meal\_type\_tags\_master(id)`.

- `dietary\_option\_tag\_ids` (elements `INTEGER`) to `dietary\_option\_tags\_master(id)`.

- `payment\_method\_tag\_ids` (elements `INTEGER`) to `payment\_methods\_master(id)`.

- Integrity for all array FKs (ensuring each referenced ID exists in its master table and that the master record has `is\_active = true`) is enforced by dedicated database triggers (`trigger\_check\_fwsrc\_serves\_meal\_types`, `trigger\_check\_fwsrc\_dietary\_options`, `trigger\_check\_fwsrc\_payment\_methods`).

- Linking Tables: `public.food\_water\_source\_media` (Version 1.0) links this table to `public.media` for managing galleries or specific media roles associated with a food/water source. Its own DDL includes audit columns, `updated\_at` trigger, and orphan translation cleanup for its translatable fields.

6\. Multilingual Strategy

- Directly Translatable Fields (Primary reference language: English, translated via `public.translations`):

- `water\_source\_access\_notes`

- Elements within `highlighted\_dishes\_local\_specialties` (each string in the array is individually translatable)

- `opening\_hours\_text\_notes`

- `specific\_notes\_for\_pilgrims`

- `food\_water\_source\_media.caption\_override` (for linked media)

- `food\_water\_source\_media.alt\_text\_override` (for linked media)

- Indirectly Translatable Fields (via `label` fields in referenced master tables, which are fetched by views or application logic):

- `source\_type\_id` (links to `food\_water\_source\_types\_master.label`)

- `water\_reliability\_id` (links to `water\_reliability\_types\_master.label`)

- `establishment\_price\_range\_id` (links to `establishment\_price\_ranges\_master.label`)

- `serves\_meal\_type\_tag\_ids` (elements link to `meal\_type\_tags\_master.label`)

- `dietary\_option\_tag\_ids` (elements link to `dietary\_option\_tags\_master.label`)

- `payment\_method\_tag\_ids` (elements link to `payment\_methods\_master.label`)

- Orphan Cleanup: The `trigger\_cleanup\_food\_water\_sources\_details\_translations` (`AFTER DELETE` on `food\_water\_sources\_details`) removes related entries from `public.translations` for this table and its media linking table (`food\_water\_source\_media`). Referenced master tables are responsible for their own translation cleanup via their respective triggers.

7\. Role-Based Workflow & RLS Notes

- Key Fields for Workflow: `data\_last\_verified\_at` and `opening\_hours\_last\_verified\_at` are crucial for content managers to track and indicate the freshness of the information. Audit columns provide traceability of changes. The overall visibility is controlled by the parent `waypoints` record's status.

- RLS Policies:

- Row-Level Security is enabled on `food\_water\_sources\_details` and its media linking table.

- Public Users (`SELECT`): Can read details of food/water sources associated with waypoints that are published (via `waypoints.content\_visibility\_status\_id` joining to `content\_statuses\_master.is\_publicly\_visible = TRUE`) and not soft-deleted (`waypoints.deleted\_at IS NULL`).

- Privileged Users (e.g., `platform\_admin`, `regional\_content\_manager` roles, identified via `public.has\_role\_on\_profile` helper): Have `ALL` permissions (CRUD).

- `WITH CHECK` conditions prevent non-`platform\_admin` users from re-parenting `waypoint\_id` on `UPDATE`.

- RLS policies for all prerequisite master tables ensure public read access only to `is\_active = true` records, while `platform\_admin` users have full control.

8\. ENUM vs Lookup Discussion

- All relevant former ENUM concepts for this table's context (`food\_water\_source\_type\_enum`, `water\_reliability\_enum`, price range ENUM, meal types ENUM, dietary options ENUM, payment methods ENUM) have been successfully promoted to full V2-compliant master lookup tables (`food\_water\_source\_types\_master` (V1.2.1), `water\_reliability\_types\_master` (V1.2.1), `establishment\_price\_ranges\_master` (V1.1), `meal\_type\_tags\_master` (V1.1), `dietary\_option\_tags\_master` (V1.1), `payment\_methods\_master` (V1.1)). This is a robust and flexible approach, providing better data integrity, translatability, lifecycle management (`is\_active`), auditability, and the ability to add descriptive attributes like icons.

9\. UI/UX Enablement

- Filtering and Identification: All `\*\_id` and `\*\_tag\_ids` fields, by linking to master tables that store `icon\_identifier` and translatable `label` fields, will power UI filters and provide clear visual cues (icons, names) for users to identify and select different types of food sources, water reliability, price ranges, meal types, dietary options, and payment methods.

- Critical Information Display: `is\_potable\_water\_source` is a key boolean flag that must be prominently and clearly displayed for water sources. `opening\_hours\_structured` can be parsed to show "Open Now?" indicators or detailed daily schedules.

- Rich Details: Fields like `highlighted\_dishes\_local\_specialties`, `water\_source\_access\_notes`, and `specific\_notes\_for\_pilgrims` provide valuable, context-specific information directly displayable on detail pages.

- Media: The `food\_water\_source\_media` table allows for a gallery, enhancing visual appeal and information.

10\. Key Considerations & Definitions

- UI Adaptation Based on `source\_type\_id`: The user interface should dynamically adapt based on the `food\_water\_source\_types\_master.is\_commercial` flag associated with the linked `source\_type\_id`. For example, fields like `establishment\_price\_range\_id`, `serves\_meal\_type\_tag\_ids`, and `payment\_method\_tag\_ids` are typically relevant only for commercial types and might be hidden or disabled for non-commercial sources like public fountains.

- Clarity on `is\_potable\_water\_source`: This field is `NOT NULL DEFAULT FALSE`. The UI must clearly differentiate for the user between sources explicitly marked as "potable" (`TRUE`), those explicitly "not potable" (`FALSE` unless other indicators like a "non-potable" `source\_type\_id` apply), and potentially "unknown/unverified" status (which might be inferred if `source\_type\_id` is a water type but `is\_potable\_water\_source` is `FALSE` and `water\_reliability\_id` indicates unknown reliability). Data entry workflows must enforce explicit confirmation for setting this to `TRUE`.

- Array FK Validation: The database triggers for `serves\_meal\_type\_tag\_ids`, `dietary\_option\_tag\_ids`, and `payment\_method\_tag\_ids` are critical for ensuring data integrity and that only active master records are referenced.

- `opening\_hours\_structured` (JSONB): Adherence to the centrally defined JSON schema for this field is critical for consistent data entry by content managers and reliable parsing by the application for display.

- Data Verification: Workflows for content managers to regularly update `opening\_hours\_last\_verified\_at` and `data\_last\_verified\_at` are important for maintaining user trust in the accuracy of the information.

11\. Scalability & Future-Proofing

- The use of structured data via linked master tables for classifications and `JSONB` for flexible data like opening hours provides excellent scalability and makes the system adaptable to future enhancements without requiring significant schema alterations to this detail table.

- Comprehensive audit columns and `is\_active` flags on master tables support robust data governance and lifecycle management.

- The 1:1 relationship with `waypoints` means that any partitioning strategy implemented for `waypoints` would naturally apply to `food\_water\_sources\_details` as well.

12\. Next-Action Checklist

- 🔴 Verify Prerequisite Master Tables are V2 Compliant:

- Confirm `public.food\_water\_source\_types\_master` (Version 1.2.1) and `public.water\_reliability\_types\_master` (Version 1.2.1) are implemented as specified.

- Confirm `public.establishment\_price\_ranges\_master` (Version 1.1), `public.meal\_type\_tags\_master` (Version 1.1), `public.dietary\_option\_tags\_master` (Version 1.1), and `public.payment\_methods\_master` (Version 1.1) DDLs (including `is\_active`, `label`, full audit, all triggers, indexes, RLS) are complete and implemented. Ensure they are populated with seed data.

- Verify `public.media\_roles\_master` is populated for use by `food\_water\_source\_media`.

- 🔴 Implement/Update `food\_water\_sources\_details` Table: Execute the DDL for Version 1.3.1.

- 🔴 Implement/Verify Triggers on `food\_water\_sources\_details`:

- `trigger\_food\_water\_sources\_details\_set\_updated\_at`.

- `trigger\_cleanup\_food\_water\_sources\_details\_translations` (including linked media).

- Active check triggers for single FKs: `trigger\_check\_fwsrc\_source\_type\_active`, `trigger\_check\_fwsrc\_water\_reliability\_active`, `trigger\_check\_fwsrc\_price\_range\_active`.

- Array FK integrity triggers (checking existence and `is\_active`): `trigger\_check\_fwsrc\_serves\_meal\_types`, `trigger\_check\_fwsrc\_dietary\_options`, `trigger\_check\_fwsrc\_payment\_methods`.

- 🔴 Implement/Verify `food\_water\_source\_media` Table: Ensure its DDL (Version 1.0), triggers (`updated\_at`, orphan translation for its translatable fields), indexes, and RLS policies are correctly implemented.

- 🟠 JSON Schema for `opening\_hours\_structured`: Ensure this schema is formally defined, documented, and consistently used.

- 🟠 RLS Helper Functions: Verify `public.has\_role\_on\_profile(UUID, TEXT)` (or equivalent) is defined and secure for use in RLS policies.

- 🟢 RLS Policies: Thoroughly test all implemented RLS policies for `food\_water\_sources\_details`, its media table, and all related master tables.

- 🟢 Translation Entries & Seed Data: Prepare initial English entries in `public.translations` for all translatable fields.

- 🟢 Data Migration (if applicable): If `food\_water\_sources\_details` already contains data, ensure schema changes are compatible.