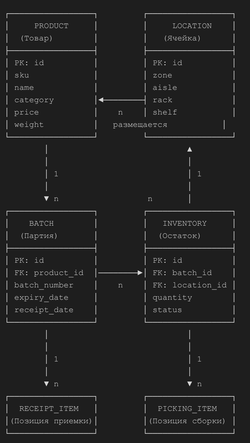
📊 Модель данных SmartWMS

Логическая и физическая модели данных, описание сущностей и связей

Версия: 1.2 | Дата: 28.03.2026 | Автор: DBA Team

ЧАСТЬ 1: ЛОГИЧЕСКАЯ МОДЕЛЬ ДАННЫХ

1.1 Концептуальная ER-диаграмма (Chen нотация)



1.2 Описание сущностей (логический уровень)

Товар (Product)

id	Уникальный идентификатор	NOT NULL PK	Внутренний ID товара
sku	Строка (50)	NOT NULL UK	Артикул товара (уникальный)
name	Строка (255)	NOT NULL	Наименование товара
category_id	Ссылка	NOT NULL FK	Категория товара
price	Decimal(10,2)	NULL	Цена (для оценки)
weight	Decimal(10,3)	NULL	Вес в кг
dimensions	JSON	NULL	Габариты {l,w,h}

Ячейка хранения (Location)

id	Уникальный идентификатор	NOT NULL PK	ID ячейки
zone	Строка (20)	NOT NULL	Зона (A, B, C...)
aisle	Строка (10)	NOT NULL	Стеллаж/ряд
rack	Строка (10)	NOT NULL	Секция
shelf	Строка (10)	NOT NULL	Полка/ячейка
barcode	Строка (50)	NOT NULL	Штрих-код ячейки
max_weight	Decimal(10,2)	NULL	Макс. вес
is_active	Boolean	NOT NULL	Активна/заблокирована

Партия (Batch)

id	Уникальный идентификатор	NOT NULL PK	ID партии
product_id	Ссылка	NOT NULL FK	Товар
batch_number	Строка (50)	NOT NULL	Номер партии
expiry_date	Date	NULL	Срок годности
receipt_date	DateTime	NOT NULL	Дата поступления
supplier_id	Ссылка	NOT NULL FK	Поставщик

ЧАСТЬ 2: ФИЗИЧЕСКАЯ МОДЕЛЬ ДАННЫХ

2.1 DDL (PostgreSQL 15)

```
-- Таблица категорий CREATE TABLE categories ( id SERIAL PRIMARY KEY, name VARCHAR(100) NOT NULL, parent_id INTEGER REFERENCES categories(id), created_at TIMESTAMP DEFAULT NOW() ); -- Таблица товаров CREATE TABLE products ( id BIGSERIAL PRIMARY KEY, sku VARCHAR(50) UNIQUE NOT NULL, name VARCHAR(255) NOT NULL, category_id INTEGER NOT NULL REFERENCES categories(id), price DECIMAL(10,2), weight DECIMAL(10,3), dimensions JSONB, is_weighted BOOLEAN DEFAULT FALSE, created_at TIMESTAMP DEFAULT NOW(), updated_at TIMESTAMP DEFAULT NOW() ); -- Индексы для поиска CREATE INDEX idx_products_sku ON products(sku); CREATE INDEX idx_products_name ON products(name); CREATE INDEX idx_products_category ON products(category_id); -- Таблица ячеек хранения CREATE TABLE locations ( id BIGSERIAL PRIMARY KEY, zone VARCHAR(20) NOT NULL, aisle VARCHAR(10) NOT NULL, rack VARCHAR(10) NOT NULL, shelf VARCHAR(10) NOT NULL, barcode VARCHAR(50) UNIQUE NOT NULL, max_weight DECIMAL(10,2), max_volume DECIMAL(10,2), location_type VARCHAR(20) -- pallet, shelf, bin is_active BOOLEAN DEFAULT TRUE, created_at TIMESTAMP DEFAULT NOW(), UNIQUE(zone, aisle, rack, shelf) ); -- Таблица партий CREATE TABLE batches ( id BIGSERIAL PRIMARY KEY, product_id BIGINT NOT NULL REFERENCES products(id), batch_number VARCHAR(50) NOT NULL, expiry_date DATE, receipt_date TIMESTAMP NOT NULL, supplier_id INTEGER REFERENCES suppliers(id), quality_status VARCHAR(20) DEFAULT 'OK', -- OK, HOLD, QUARANTINE created_at TIMESTAMP DEFAULT NOW(), UNIQUE(product_id, batch_number) ); -- Таблица остатков (партия + ячейка) CREATE TABLE inventory ( id BIGSERIAL PRIMARY KEY, batch_id BIGINT NOT NULL REFERENCES batches(id), location_id BIGINT NOT NULL REFERENCES locations(id), quantity DECIMAL(15,3) NOT NULL CHECK (quantity >= 0), reserved_quantity DECIMAL(15,3) DEFAULT 0, status VARCHAR(20) DEFAULT 'AVAILABLE', -- AVAILABLE, RESERVED, QUARANTINE last_count_date TIMESTAMP, created_at TIMESTAMP DEFAULT NOW(), updated_at TIMESTAMP DEFAULT NOW(), UNIQUE(batch_id, location_id) ); -- Индексы для быстрого поиска CREATE INDEX idx_inventory_location ON inventory(location_id); CREATE INDEX idx_inventory_batch ON inventory(batch_id); CREATE INDEX idx_inventory_expiry ON batches(expiry_date) WHERE expiry_date IS NOT NULL; -- Таблица документов приемки CREATE TABLE receipts ( id BIGSERIAL PRIMARY KEY, document_number VARCHAR(50) UNIQUE NOT NULL, supplier_id INTEGER REFERENCES suppliers(id), receipt_date TIMESTAMP NOT NULL, status VARCHAR(20) DEFAULT 'EXPECTED', -- EXPECTED, IN_PROGRESS, COMPLETED created_by INTEGER REFERENCES users(id), created_at TIMESTAMP DEFAULT NOW() ); -- Партиционирование по дате (для больших таблиц) CREATE TABLE receipt_items ( id BIGSERIAL, receipt_id BIGINT NOT NULL REFERENCES receipts(id), product_id BIGINT NOT NULL REFERENCES products(id), expected_quantity DECIMAL(15,3) NOT NULL, received_quantity DECIMAL(15,3) DEFAULT 0, batch_id BIGINT REFERENCES batches(id), location_id BIGINT REFERENCES locations(id), created_at TIMESTAMP DEFAULT NOW() ) PARTITION BY RANGE (created_at);
```

2.2 Индексы и производительность

```
-- Составные индексы для частых запросов CREATE INDEX idx_inventory_location_status ON inventory(location_id, status); CREATE INDEX idx_batches_product_expiry ON batches(product_id, expiry_date); -- Full-text поиск по товарам CREATE INDEX idx_products_fts ON products USING GIN (to_tsvector('russian', name)); -- Индекс для геозон (если используем PostGIS) -- CREATE INDEX idx_locations_geom ON locations USING GIST (geom); -- Индекс для JSON полей CREATE INDEX idx_products_dimensions ON products USING GIN (dimensions);
```

2.3 Словарь данных (Data Dictionary)

Таблица: inventory (остатки)

id	bigserial	NOT NULL PK	Суррогатный ключ
batch_id	bigint	NOT NULL FK → batches.id	Ссылка на партию
location_id	bigint	NOT NULL FK → locations.id	Ссылка на ячейку
quantity	decimal(15,3)	NOT NULL	Текущее количество
reserved_quantity	decimal(15,3)	DEFAULT 0	Зарезервировано под заказы
status	varchar(20)	NOT NULL	AVAILABLE/RESERVED/QUARANTINE

ЧАСТЬ 3: СВЯЗИ МЕЖДУ СУЩНОСТЯМИ

Product (1) ----- (n) Batch
Один товар может иметь множество партий (поставок)

Batch (1) ----- (n) Inventory
Одна партия может быть разложена в несколько ячеек

Location (1) ----- (n) Inventory
В одной ячейке могут храниться разные партии

Receipt (1) ----- (n) ReceiptItem
Один документ приемки содержит много позиций

Order (1) ----- (n) PickingItem
Один заказ может собираться из нескольких ячеек

ЧАСТЬ 4: ТИПОВЫЕ ЗАПРОСЫ

```
-- Найти все ячейки с просроченным товаром SELECT l.zone, l.aisle, l.rack, l.shelf, p.name, b.batch_number, b.expiry_date, i.quantity FROM inventory i JOIN batches b ON i.batch_id = b.id JOIN products p ON b.product_id = p.id JOIN locations l ON i.location_id = l.id WHERE b.expiry_date < CURRENT_DATE AND i.quantity > 0; -- Рекомендация ячейки для размещения (свободные ячейки в зоне A) SELECT * FROM locations WHERE zone = 'A' AND is_active = TRUE AND id NOT IN ( SELECT location_id FROM inventory WHERE quantity > 0 ) LIMIT 1; -- Оборачиваемость товаров (топ-10) SELECT p.sku, p.name, SUM(oi.quantity) as total_sold FROM order_items oi JOIN products p ON oi.product_id = p.id WHERE oi.created_at > NOW() - INTERVAL '30 days' GROUP BY p.id, p.sku, p.name ORDER BY total_sold DESC LIMIT 10;
```

Версии модели:

- v1.0 - 20.03.2026 - Базовая структура (товары, ячейки)
- v1.1 - 25.03.2026 - Добавлены партии и остатки
- v1.2 - 28.03.2026 - Оптимизация индексов, партиционирование