

3D生成网站数据表设计与数据字典（v0.5）

目标：支撑“根据文字/图片调用第三方大模型生成3D模型”的站点，形成可上线的关系型数据底座与数据字典。以 PostgreSQL 为基准（兼容 MySQL 需做少量调整）。本版在 v0.1 基础上补齐计费/权限/分享/限流/数据血缘/后处理与发布等关键环节。

- 版本：v0.5（2025-09-26）
- 范围：核心交易域（Prompt→Job→Asset）、运营域（评估/发布/分享）、平台域（鉴权/计费/限流/审计）

A. 变更摘要（v0.5 对比 v0.1）

1) 表扩展：assets、generation_jobs、provider_models、model_assets、evaluations 增补关键字段（见各表）。 2) 新增表：api_keys、rate_limits、share_links、conversions、datasets/dataset_items、moderation_events、notifications。 3) 状态机落地：明确 Job/Conversion 的状态流转与重试关系（含 retry_of_job_id）。 4) 计费与用量：在 Job/Request 层增加 billed_units_json 与 cost_usd 细分来源；保留对账与发票扩展位。 5) 权限与分享：项目内 RBAC 继续沿用，补充外链分享 share_links（一次性/限期/限次数）。 6) 可观察性：统一请求日志、评估指标、质量元数据与数据血缘（资产来源、转换链）。

0. 设计原则与通用约定

- 命名：表名复数 snake_case；主键 id（UUID）；外键以 _id 结尾。
- 通用字段：created_at timestampz，updated_at timestampz，created_by uuid，updated_by uuid；软删（可选）deleted_at timestampz。
- 多租户：所有业务表包含 org_id；跨组织访问一律禁止。
- 文件存储：仅存元数据与校验摘要；大文件走对象存储。
- JSON 使用：对不稳定/供应商差异化字段用 jsonb；沉淀稳定后再强类型化。
- 审计与可追溯：关键写操作入 audit_logs；资产保留来源链与转换链。

1. 核心实体关系（文字版 ER）

Org → Project → Prompt → GenerationJob → ModelAsset/Asset → Scene - GenerationJob 调用 ProviderModel，产出主 ModelAsset 与附属 Asset(贴图/预览)；过程有若干 ProviderRequest；后处理记录在 conversions。 - evaluations 绑定在 job 或 model_asset 上；moderation_events 记录合规模块。 - 对外分享使用 share_links；外部/自动化访问用 api_keys 并受 rate_limits 限制。

2. 最小可行表（MVP + v0.5 扩展）

2.1 orgs（组织）

id, name, plan, status, created_at, updated_at

2.2 users (用户)

id, org_id, email, name, role, status, metadata, created_at, updated_at

2.3 projects (项目)

id, org_id, name, description, visibility, created_by, updated_by, created_at, updated_at

2.4 project_members (项目成员)

id, project_id, user_id, role, created_at

2.5 assets (通用文件资产, 含输入与产出)

- 新增/调整字段:
- visibility text (private/team/public)
- source_type text (upload/generated/imported)
- origin_url text (外部来源, 可为空)
- license text (版权/许可证标记)
- sha256 text (去重摘要) + md5 text (快速校验, 可选)
- storage_region text、etag text、signed_url_expires_at timestampz
- lifecycle_state text (active/archived/purged) 与 retention_until timestampz
- 字段总览 (节选): id, org_id, project_id, owner_id, kind, name, storage_provider, storage_bucket, storage_key, storage_url, sha256, md5, mime_type, size_bytes, width, height, duration_sec, source_type, origin_url, license, visibility, storage_region, etag, signed_url_expires_at, lifecycle_state, retention_until, metadata jsonb, created_at, updated_at

2.6 prompts (文本/图片提示词)

- 新增/调整: version int default 1、translated_text jsonb (多语)、safety_tags text[]。
- 字段总览: id, org_id, project_id, user_id, input_type, text, input_asset_id, negative_text, parameters jsonb, prompt_hash, language, translated_text, moderation_status, safety_tags, tags text[], version, created_by, created_at

2.7 providers & provider_models (供应商与模型)

- providers(id, name, base_url, status, created_at)
- provider_models(id, provider_id, model_name, task_type, pricing jsonb, limits jsonb, metadata jsonb, status, created_at)
- 新增: limits jsonb (最大面数/文件体积/并发/速率), 用于前置校验与排错。

2.8 generation_jobs (生成任务)

- 新增/调整字段:
- 调度/重试: queue_id uuid、attempt int、retry_of_job_id uuid、retried_by_job_id uuid、scheduler_lane text、worker_id text、worker_region text
- 性能与计费: queue_latency_ms int、billed_units_json jsonb (如{"task":1,"mesh_mb":18})
- 生命周期: auto_cleanup_at timestampz
- 字段总览: id, org_id, project_id, prompt_id, provider_id, provider_model_id, status, priority, queue_id, attempt, retry_of_job_id, retried_by_job_id, scheduler_lane, worker_id, worker_region, started_at, completed_at, latency_ms, queue_latency_ms,

cost_usd, billed_units_json, error_code, error_message, webhook_url, created_by, created_at

2.9 provider_requests (底层API往返日志)

id, job_id, request_ts, response_ts, request_payload_jsonb, response_payload_jsonb, http_status, tokens_input, tokens_output, billed_cost_usd, retry_index int, provider_request_id text

2.10 model_assets (生成的3D主资产)

- 新增/调整: lod_of_model_asset_id uuid (本记录为某模型的降阶/简化版本)、uv_map_count int、triangulated boolean、watertight boolean
- 字段总览: id, job_id, asset_id, format, vertex_count, face_count, material_count, uv_map_count, bbox_jsonb, units, origin, triangulated, watertight, preview_asset_id, lod_of_model_asset_id, quality_metrics_jsonb, created_at

2.11 model_textures (模型-贴图 关联)

id, model_asset_id, texture_asset_id, role, uv_channel

2.12 scenes / scene_items (场景装配)

- scenes(id, project_id, name, description, created_at, updated_at)
- scene_items(id, scene_id, model_asset_id, transform_jsonb, created_at)

2.13 evaluations (自动/人工评估)

- 新增: rubric_id text (评估量表版本, 保证可比性)
- 字段: id, subject_type, subject_id, evaluator_type, rubric_id, metrics_jsonb, comment, rating, created_by, created_at

3. 新增平台/运营表 (v0.5)

3.1 api_keys (访问密钥)

id, org_id, name, key_hash, scopes text[], last_used_at, expires_at, status, created_at

3.2 rate_limits (限流配额)

id, org_id, key_id, window text, max_requests int, max_concurrency int, used_in_window int, window_started_at, updated_at

3.3 share_links (外链分享)

id, org_id, subject_type, subject_id, token_hash, visibility text(private/unlisted/public), expires_at, max_accesses int, access_count int, created_by, created_at

3.4 conversions (后处理/转换任务)

- 用于记录 glb→usdz、网格简化、贴图烘焙、预览渲染等。

- 字段: `id, org_id, source_asset_id, operation text, parameters jsonb, status text(pending/running/succeeded/failed), started_at, completed_at, output_asset_id, error_code, error_message, created_at`

3.5 datasets / dataset_items (数据集与条目)

- 采集/标注/训练前的数据资产打包与血缘追踪。
- `datasets(id, org_id, name, description, purpose text(train/eval/benchmark), created_by, created_at)`
- `dataset_items(id, dataset_id, asset_id, labels jsonb, split text(train/val/test), created_at)`

3.6 moderation_events (合规与内容审核)

`id, org_id, subject_type(job/prompt/asset), subject_id, rule text, severity text, details jsonb, decision text(allow/flag/block), created_at`

3.7 notifications (站内通知/事件)

`id, org_id, user_id, event_type, payload jsonb, read boolean, created_at`

4. 枚举与取值建议 (更新)

- `assets.kind`: `file | image | thumbnail | 3d_model | texture | material`
- `assets.source_type`: `upload | generated | imported`
- `prompts.input_type`: `text | image | both`
- `generation_jobs.status`: `pending | queued | running | succeeded | failed | canceled`
- `conversions.status`: 同上
- `evaluations.evaluator_type`: `auto | human`
- `share_links.visibility`: `private | unlisted | public`

5. 状态机与流转 (关键)

GenerationJob: `pending → queued → running → (succeeded | failed)`; failed 可基于策略入队重试 (attempt++, 写 `retry_of_job_id`)

Conversions: `pending → running → (succeeded | failed)`; 输出 `output_asset_id` 形成**转换链** (资产血缘 = 输入链 + 转换链)。

失败治理: `provider_requests.http_status != 200` 或解析异常 → `generation_jobs.error_code` 归因 (TIMEOUT/RATE_LIMIT/PROVIDER_ERROR/PAYLOAD_TOO_LARGE...)。

6. 索引、分区与性能

- 高频筛选: B-Tree on `org_id, project_id, status, provider_model_id, created_at`。
- 去重: `assets(sha256) + UNIQUE(storage_key)`; 大表 (月分区) `generation_jobs, provider_requests, conversions`。

- 文本检索: `prompts(text)` 建立全文索引 (TSVECTOR) ; 向量检索用 `embeddings` (pgvector) 。
- JSONB 查询热点加 GIN: 如 `provider_requests(response_payload)`、`evaluations(metrics)` 。
- 外键建议 `DEFERRABLE INITIALLY DEFERRED` 以批量导入友好; 大事务控制 `statement_timeout` 与批量大小。

7. 数据治理与合规

- 权限: 项目内 RBAC; API Key 以最小权限 + 过期时间; 下载 3D 源仅 `editor+` 。
- 隐私: 请求/响应日志最小必要保存; 脱敏个人信息; 分享链接默认 `unlisted` 且限期。
- 保留策略: 失败 Job/Request/Moderation 默认 90 天; 资产可按配额或 `retention_until` 清理。
- 审计: 删除/公开发布/下载/生成调用等写入 `audit_logs` ; 可关联请求 `id` 串联链路。

8. 典型流程 (更新版)

1) 用户提交 `Prompt` (可能含参考图 `Asset`) 。 2) 生成 `GenerationJob` → 入队; 产生 1..n 次 `ProviderRequest` (含重试) 。 3) 成功落地主 3D `Asset(kind=3d_model)` + `ModelAsset` ; 贴图 `Asset(kind=texture)` , 预览 `Asset(kind=thumbnail)` 。 4) 进入 `conversions` (如简化/格式转换/预览渲染) , 产出派生 `Asset` 并记录血缘 (`source_asset_id` → `output_asset_id`) 。 5) 自动/人工 `evaluations` ; 如命中审核规则, 登记 `moderation_events` 。 6) 可创建 `share_links` 对外阅览或下载; 若为自动化访问, 使用 `api_keys` 并受 `rate_limits` 控制。

9. 附: DDL 样例 (PostgreSQL)

仅示例两表, 其余可按同风格生成。

```
-- assets (节选)
create table assets (
  id uuid primary key,
  org_id uuid not null,
  project_id uuid,
  owner_id uuid,
  kind text not null,
  name text,
  storage_provider text not null,
  storage_bucket text not null,
  storage_key text not null unique,
  storage_url text,
  sha256 text,
  md5 text,
  mime_type text,
  size_bytes bigint,
  width int,
  height int,
  duration_sec numeric,
  source_type text,
  origin_url text,
```

```

license text,
visibility text default 'private',
storage_region text,
etag text,
signed_url_expires_at timestampz,
lifecycle_state text default 'active',
retention_until timestampz,
metadata jsonb default '{} '::jsonb,
created_at timestampz not null default now(),
updated_at timestampz not null default now()
);
create index on assets (org_id, project_id);
create index on assets (kind);
create index on assets (sha256);

-- generation_jobs (节选)
create table generation_jobs (
  id uuid primary key,
  org_id uuid not null,
  project_id uuid,
  prompt_id uuid not null,
  provider_id uuid,
  provider_model_id uuid,
  status text not null,
  priority int default 5,
  queue_id uuid,
  attempt int default 0,
  retry_of_job_id uuid,
  retried_by_job_id uuid,
  scheduler_lane text,
  worker_id text,
  worker_region text,
  started_at timestampz,
  completed_at timestampz,
  latency_ms int,
  queue_latency_ms int,
  cost_usd numeric(12,6),
  billed_units_json jsonb default '{} '::jsonb,
  error_code text,
  error_message text,
  webhook_url text,
  created_by uuid,
  created_at timestampz not null default now()
);
create index on generation_jobs (org_id, project_id, status);
create index on generation_jobs (provider_model_id);
create index on generation_jobs (created_at);

```

10. 交付与后续工作

- 如需**完整 DDL**（全部表）或 **MySQL 版本**，可以在本版基础上直接生成。
- 如需 **数据样例 + BI 需求对齐**，可导出字段清单与口径说明表。
- 若启用检索/相似度，建议落地 `embeddings`（pgvector）并统一文本归一化策略。