# 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 timestamptz，updated\_at timestamptz，created\_by uuid，updated\_by uuid；软删（可选）deleted\_at timestamptz。
* **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 timestamptz
  + lifecycle\_state text（active/archived/purged）与 retention\_until timestamptz
* **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 timestamptz
* **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）

-- 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 timestamptz,  
 lifecycle\_state text default 'active',  
 retention\_until timestamptz,  
 metadata jsonb default '{}'::jsonb,  
 created\_at timestamptz not null default now(),  
 updated\_at timestamptz 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 timestamptz,  
 completed\_at timestamptz,  
 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 timestamptz 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);