



dcc-mcp-PowerPoint 能力边界 (v2)



dcc-mcp-PowerPoint 能力边界 (v2)

补丁引擎实测

- 本页由 add_slide 动态插入
- 标题由 set_text 改写 (第 1 页)
- 备注由 set_notes 改写 (第 17 页)
- 强调色由 set_shape_fill 换为暖橙 (第 2 页)

01

生成管线

Deck IR → Open XML 编译 → COM 渲染 → 结构验证

生成管线四步

- Deck IR 契约 (office-ir/1.0) : 语义布局 + 内容块, 绝不手写坐标
- Open XML 编译器: 11 种布局注册表, 一键扩展, 不碰分发逻辑
- 桌面 COM 渲染: PDF + 逐页 PNG (DispatchEx 专用实例)
- 结构验证: 信封契约 + 产物存在性 + 溢出与对比度检查

架构分层



Deck IR: 镜像 dcc-mcp-office-ir 演示文稿 schema

DeckCompiler: 语义布局注册表 → python-pptx Open XML

Renderer: PowerPoint COM → PDF / PNG 预览

Validate + Analyze: 结构与视觉问题诊断

Layers / Edits / Plugins: 生成后能力面 (本分支新增)

能力数字

11
语义布局

6
内置图层

12
补丁操作

4
技能包

2
渲染产物 (PDF/PNG
)

0
运行时第三方依赖

双后端对比

Open XML 后端

- 无 Office 环境可编译
- 结构确定性、可版本管理
- 毫秒级生成

COM 渲染后端

- PDF 矢量导出
- 逐页 PNG 预览
- Office 缺席时显式降级

设计取舍

结构化优先

- ✓ 语义布局而非坐标
- ✓ 契约先行、验证兜底
- ✓ 写操作不静默降级

能力边界内自研

- ✓ C# 宿主承重 (COM/OpenXML)
- ✓ 越界能力回上游 dcc-mcp-office
- ✓ 缺陷修复优先提 PR 上游

一次调用的旅程



真实案例：Blender Lookdev



- 来自 DCC 渲染管线的真实产物
- 图片左置 + 要点右侧面板
- alt 文本随图片写入（可访问性）
- 缺失资产显式占位，绝不静默丢图

覆盖的 DCC 工具



Maya



Blender



Houdini



Photoshop



Unity



Unreal



Nuke



3ds Max

02

语义图层

原生 OOXML 标签 `::layer=` 存活于 PowerPoint 往返

六个内置图层

- background 整页背景 · decoration 幽灵圆/logo/分隔线

-
- header 标题栏 · content 要点/卡片/图片

-
- accent 强调条/圆点/连接线 · footer 页脚/页码

-
- 图层操作：显隐、z 序重排、重着色、形状归属

图层操作面

1
layer_list 清点

2
set_visibility 显隐

3
reorder 前后/上下

4
recolor 重着色

5
assign 归属

6
按幻灯片范围

03

补丁引擎与插件

ppt-patch/1.0 全有或全无 · plugin.json 隔离子进程

补丁操作面

- set_text / set_notes: 文本与备注替换
- set_image / set_shape_fill / set_shape_line: 媒体与样式
- set_shape_visible / delete_shape: 显隐与删除
- add_slide / delete_slide / move_slide: 页面结构
- assign_layer: 形状归属图层
- 选择器无坐标: slide + layer + shape + id + index

插件注册表

- plugin.json 契约: name/version/script/interpreter/schema
- 显式发现根: DCC_POWERPOINT_PLUGIN_PATH + 用户目录
- 隔离子进程: stdin JSON → stdout 单 JSON 结果
- 永不自动下载、永不 import 插件代码



Thanks

dcc-mcp-PowerPoint · 让 AI 做高品质幻灯片