Council Tools — `export_document` Implementation

Date: 2026-11-03 · **Host:** ai1 · **Status:** Live & verified end-to-end

Summary

Added a single thin MCP tool, `export_document`, to the council-tools server (`oc_mcp.py`). It proxies to the existing `doc-api` service (port 5051), which already builds PDF, DOCX, and XLSX from markdown. No new export logic and no changes to `notify_api.py`.

Tool signature

```
@mcp.tool()
def export_document(content: str, title: str, format: str = "pdf",
                    post_to_channel: bool = True,
                    user_id: str = None, ctx: Context = None) -> str:
    ...
```

• `format` ∈ {`pdf`, `docx`, `xlsx`} → maps to `doc-api /generate?type=`
• Returns the download URL: `https://docs.ai1.dns-1.net/?download=1`
• When `post_to_channel=true`, also posts the link to the caller's OC channel via the same `_caller_id(ctx)` routing used by `generate_image`.
A new constant was added: `DOC_URL = "http://localhost:5051"`.

Files changed

File	Change
`/mnt/nvme/council/oc_mcp.py`	Added `DOC_URL` + `export_document` tool
`/home/owner/install_council.sh`	Heredoc updated (constant, tool, port-substitution sed line)

Verification

• `bash -n install_council.sh` → OK; regenerated heredoc compiles; `DOC_API_PORT_PLACEHOLDER` resolves correctly.
• `council-mcp.service` restarted and active.
• `tools/list` over :8765 now includes `export_document`.
• Live call returned a real file: `Roundtrip_Check_20261103_201507.xlsx` with working download URL.

Notes

Coder and Halogen already have the `council` tool-server attached, so the new tool is available to them automatically — no per-model change required.