# Workflow: Canvas Versioning via File Uploads

This workflow ensures you can iterate on Canvases without running into context window limits or losing visibility. It converts fragile, conversation-bound Canvases into durable, project-wide Files.

## Why This Workflow

* **Conversations are siloed**: A Canvas only exists in the thread where it was created.
* **Files are project-wide**: Once uploaded, any conversation in the same Project can access them.
* **Context windows are finite**: Resetting or bloating a thread severs Canvas links. Exporting keeps your work portable.

This system makes each Canvas version portable, searchable, and safe from context overflow.

## Step-by-Step Process

### 1. Work in a Canvas (in a conversation)

* Draft, edit, and refine your Canvas until it reaches a stable point.
* Before exporting, add a brief **Change Log** section noting what was updated.

### 2. Export the Canvas

* Download the Canvas as a file (.docx or .md recommended).
* Name it with a clear version label, e.g.:
  + Branding-Setup\_v1.docx
  + Branding-Setup\_v2.docx

### 3. Upload the File to Your Project

* Drag the exported file into the Project.
* Once uploaded, it becomes globally visible across all convos in this Project.

### 4. Start a Fresh Conversation

* Open a new convo to avoid context bloat.
* Upload the **latest version file** (e.g., Branding-Setup\_v2.docx).
* Ask ChatGPT to create a new Canvas seeded with the file’s content.

### 5. Edit in the New Canvas

* Continue iterating in the new Canvas.
* When stable again, repeat the export → upload → new convo cycle.

## Key Advantages

* **Full visibility**: The file version is always accessible, even if the Canvas link breaks.
* **Context reset**: Each convo stays lightweight — no overflow.
* **Version control**: Explicit file names prevent confusion.
* **Cross-referencing**: Upload multiple files into a convo if you need Canvases to “talk to each other.”

## Pitfalls to Avoid

* ❌ **Relying only on Canvases** — they’re not visible outside their convo.
* ❌ **Losing version lineage** — always include a Change Log before exporting.
* ❌ **File bloat** — archive or move old versions out of the Project if performance slows.

## Best Practices

* Keep a simple **Version Tracking Table** (in Notion, Airtable, or CSV) with:
  + File name
  + Date exported
  + Key changes
  + Related convo link
* Use .docx or .md for easier re-editing (PDFs lock formatting).
* For hub documents (like Strategy or Branding), always cross-upload all relevant files so ChatGPT can see them together.

## Example Workflow in Action

1. Edit **1 Branding/Setup** Canvas in Convo A.
2. Export → Branding-Setup\_v1.docx.
3. Upload to Project.
4. Start Convo B → seed with Branding-Setup\_v1.docx → create new Canvas.
5. Edit → Export as Branding-Setup\_v2.docx.
6. Repeat cycle.

✅ This ensures continuity, visibility, and scalability — while dodging context overflow and duplicate Canvas issues.