

1 Page Maintenance

Steam Association of Manitoba — Maintaining and Creating Website Pages

This document describes how content is added and maintained within the SAM static website.

A “page” corresponds to a **directory** within the `/site` tree.

The site generator (`sitengen`) produces an `index.html` file for each directory based on its contents and its `metadata.txt` file.

The instructions below are divided into two parts:

- **Part A — Maintaining an existing page**
- **Part B — Creating a new page**

All procedures are written in an impersonal, third-person style for clarity and longevity.

2 Part A — Maintaining an Existing Page

Routine maintenance includes:

- changing a description
- adding a new file
- removing a file
- adjusting headings or dividers
- updating preamble or postamble HTML

These tasks modify an existing directory and its metadata.

2.1 Changing a Description

1. Open the directory’s `metadata.txt` file.
2. Locate the entry whose description needs updating.
3. Edit the description text.
4. Run the site generator (`generate-site.bat`).
5. Preview or upload the updated site as needed.

The generator will rebuild the directory’s `index.html` using the updated description.

2.2 Adding a File to an Existing Page

1. Place the new file (e.g., PDF) into the directory.
2. Run the site generator.
 - The generator will detect the new file and add a new entry to `metadata.txt` with an **empty description**.
3. Edit `metadata.txt` to add a meaningful description.
4. Run the site generator again to update `index.html`.

This two-pass process ensures metadata remains authoritative.

2.3 Removing a File from an Existing Page

1. Delete the file from the directory.
2. Run the site generator.
 - The generator will comment out the corresponding line in `metadata.txt`.
3. Optionally remove the commented line manually.

The generator never deletes user-authored metadata; it only comments out obsolete entries.

2.4 Adjusting Headings or Dividers

Headings (`*Heading:`) and dividers (`-:` or `file.pdf:` -) may be added, removed, or rearranged directly in `metadata.txt`.

After editing:

1. Run the site generator.
2. Preview or upload the updated site.

2.5 Previewing Changes

A local preview server may be started using:

```
preview-site.bat
```

This launches a simple Python HTTP server rooted at `/site`, allowing the updated pages to be viewed in a browser before uploading.

3 Part B — Creating a New Page

A new page is created by adding a new directory **within the existing** `/site` **structure**. The generator will automatically add a link to the new page in the parent directory's metadata.

Creating a new page is a two-pass process:

1. First pass: generator creates an initial `metadata.txt` with empty descriptions.
2. Human edits the metadata.
3. Second pass: generator produces the final `index.html`.

3.1 Step 1 — Create a New Directory

A new page must reside within the existing directory hierarchy.

Examples:

```
/site/public/Forms  
/site/members/BoilerSafety  
/site/members/Articles/AdvancedTopics
```

The directory name becomes part of the page URL.

3.2 Step 2 — Add Files (Optional)

Any initial documents may be placed into the new directory:

```
boiler-basics.pdf  
inspection-checklist.pdf
```

Files may also be added later; the generator will detect them.

3.3 Step 3 — First Site Generation Pass

Run:

```
generate-site.bat
```

The generator will:

- create a new `metadata.txt` file

- populate it with entries for all files and subdirectories
- leave all descriptions empty
- add a link to this new directory in the parent directory's metadata

At this point, the new page exists structurally but has no meaningful descriptions.

3.4 Step 4 — Edit `metadata.txt`

Open the newly created `metadata.txt` and add:

- descriptions for each file
- headings (`*Heading:`)
- dividers (`-:`)
- optional directory descriptions
- optional `hide` markers

Example:

```
*Boiler Safety:
boiler-basics.pdf: Introduction to boiler safety
inspection-checklist.pdf: Daily inspection checklist
```

The order of lines determines the order of rows in the final page.

3.5 Step 5 — Second Site Generation Pass

Run:

```
generate-site.bat
```

This pass:

- updates the directory's `index.html`
- updates the parent directory's `index.html`
- ensures all metadata is synchronized

The new page is now fully generated.

3.6 Step 6 — Preview the New Page

Start the local preview server:

```
preview-site.bat
```

Navigate to the new directory in a browser to confirm:

- descriptions appear correctly
 - headings and dividers are correct
 - links work
 - preamble/postamble (if present) render correctly
-

3.7 Step 7 — Upload the Updated Site

When satisfied with the preview, upload the site:

```
upload-site.bat
```

This publishes the entire `/site` directory to the server.

4 Summary

Maintaining existing pages involves editing `metadata.txt`, adding or removing files, and regenerating the site.

Creating a new page involves:

1. Creating a directory
2. Running the generator (first pass)
3. Editing metadata
4. Running the generator again (second pass)
5. Previewing
6. Uploading

The system is designed to be simple, predictable, and easy to maintain over time.