# ESP32 Open Gateway Learnings and notes

## What is the difference between spiffs and this: #include <Preferences.h>?

Great question! \*\*SPIFFS\*\* and \*\*Preferences\*\* are two completely different storage systems on the ESP32, each designed for different purposes:

## \*\*Preferences (NVS - Non-Volatile Storage)\*\*

### \*\*Purpose\*\*: Simple key-value storage for configuration data

```cpp

#include <Preferences.h>

Preferences preferences;

preferences.begin("my-app", false);

preferences.putString("wifi\_ssid", "MyWiFi");

preferences.putInt("brightness", 75);

String ssid = preferences.getString("wifi\_ssid", "default");

preferences.end();

```

### \*\*Characteristics\*\*:

- \*\*Small data\*\* - Individual values (strings, integers, floats)

- \*\*Key-value pairs\*\* - Like a simple database

- \*\*Built into ESP32\*\* - Part of the NVS system

- \*\*Automatic management\*\* - Handles wear leveling, corruption recovery

- \*\*Limited size\*\* - Typically a few KB total

- \*\*Very reliable\*\* - Designed for configuration data

### \*\*Use cases\*\*:

- WiFi credentials

- User settings (brightness, timeouts)

- Calibration values

- Device configuration

- Small persistent data

## \*\*SPIFFS (SPI Flash File System)\*\*

### \*\*Purpose\*\*: File system for storing actual files

```cpp

#include "SPIFFS.h"

SPIFFS.begin();

File file = SPIFFS.open("/config.json", "w");

file.print("{\"setting\": \"value\"}");

file.close();

File readFile = SPIFFS.open("/webpage.html", "r");

String content = readFile.readString();

readFile.close();

```

### \*\*Characteristics\*\*:

- \*\*Large files\*\* - HTML, CSS, JavaScript, images

- \*\*File system\*\* - Directories, files, file operations

- \*\*Larger storage\*\* - Can use MB of space

- \*\*File-based\*\* - Read/write entire files

- \*\*More complex\*\* - Need to manage files manually

- \*\*Web content\*\* - Perfect for serving web pages

### \*\*Use cases\*\*:

- Web interface files (HTML, CSS, JS)

- Log files

- Configuration files (JSON, XML)

- Certificates and keys

- Any file-based data

## \*\*Key Differences\*\*:

| Feature | Preferences (NVS) | SPIFFS |

|---------|-------------------|---------|

| \*\*Data Type\*\* | Key-value pairs | Files |

| \*\*Size\*\* | Small (few KB) | Large (MB) |

| \*\*Complexity\*\* | Simple | More complex |

| \*\*Use Case\*\* | Settings/Config | Web files/Logs |

| \*\*API\*\* | `putString()`, `getInt()` | `open()`, `read()`, `write()` |

| \*\*Reliability\*\* | Very high | Good |

| \*\*Management\*\* | Automatic | Manual |

## \*\*In Your Project\*\*:

### \*\*You use Preferences for\*\*:

```cpp

preferences.getString("m1\_name"); // Meter names

preferences.putString("wifi\_ssid", ssid); // WiFi settings

```

### \*\*You would use SPIFFS for\*\*:

```cpp

File file = SPIFFS.open("/index.html", "r"); // Web pages

String webpage = file.readString(); // Large HTML content

```

## \*\*Why Both Exist\*\*:

- \*\*Preferences\*\*: Perfect for small, frequently accessed configuration data

- \*\*SPIFFS\*\*: Perfect for large files that don't change often

Think of \*\*Preferences\*\* as a simple settings database, and \*\*SPIFFS\*\* as a file cabinet. You wouldn't store a large HTML file in Preferences, and you wouldn't use SPIFFS for a simple WiFi password.

## \*\*In Your Migration\*\*:

- \*\*Keep using Preferences\*\* for meter names, WiFi settings, etc.

- \*\*Use SPIFFS\*\* for the large HTML webpage content

- This gives you the best of both worlds!

Both systems are \*\*non-volatile\*\* (survive power loss), but they're optimized for completely different use cases.