

Migrating Homebridge plugin to v2.2.1

Session ID: ses_070952403ffe26OLVyPAiRDxy8 **Created:** 7/23/2026, 10:39:32 AM **Updated:** 7/23/2026, 11:29:48 AM

User

This homebridge plugin was originally written for homebridge 1.0. However due to some breaking changes in the latest Homebridge v2.2.1 upgrade, it no longer works due to different required packages. Please migrate the current code base to Homebridge v2.2.1 while avoiding any functional regressions.

Assistant

Thinking:

The user wants to migrate this homebridge plugin from v1.0 to v2.2.1. Let me first read the current codebase to understand what needs to change.

Tool: read

Tool: read

Assistant

Thinking:

Let me research what breaking changes happened in Homebridge v2.x compared to v1.0, particularly around accessory registration and HAP API.

Let me research the Homebridge v2.x breaking changes.

Tool: webfetch

Tool: webfetch

Assistant

Thinking:

Let me get the specific migration guide and the homebridge v2 API changes.

Tool: webfetch

Tool: webfetch

Assistant

Thinking:

Let me get more details on the HAP-NodeJS v1 changes and the homebridge API for accessories.

Tool: webfetch

Tool: webfetch

Assistant

Thinking:

The developer docs aren't rendering well. Let me try to fetch the raw API reference and the HAP-NodeJS changelog.

Tool: webfetch

Tool: webfetch

Assistant

Thinking:

Now I have a good understanding of the changes. Let me look at the key breaking changes for this plugin:

1. **HAP-NodeJS v1** - The HAP library is now `@homebridge/hap-nodejs` instead of being bundled with `homebridge`
2. **`Characteristic.getValue()` removed** - Replaced with `Characteristic.value` property
3. **Enums off `Characteristic` class removed** - Things like `Characteristic.Units`, `Characteristic.Formats`, `Characteristic.Perms` must now be accessed via `api.hap.Units` etc.
4. **`Accessory.getServiceByUUIDAndSubType()` removed** - Use `getServiceById()` instead
5. **`Accessory.updateReachability()` removed** - No longer supported
6. **`Accessory.setPrimaryService()` removed** - Use `Service.setPrimaryService()` instead
7. **`BatteryService` removed** - Use `Battery` instead
8. **Node.js version requirements** - Now `^22.12.0 || ^24.0.0`

For this accessory plugin specifically, I need to check what's actually being used in the code. The `homebridge.registerAccessory` call should remain compatible, and `homebridge.hap.Service` and `homebridge.hap.Characteristic` are still accessible, but I'm not seeing any `getValue()` calls that need updating.

Looking at the HTTP request handling, the code uses Node's built-in `http` module rather than the `request` package, so that's fine. The main issue is that `package.json` has `request` as an unused dependency that should be removed. The `polling-to-event` package is still being used and should stay. I should also update the `engines` field to reflect the new Node.js version requirements and check if there are any other compatibility issues with Homebridge v2.

Looking at the migration guide more carefully, the key changes are that `Characteristic.getValue()` has been replaced with direct property access, and enum constants that were previously accessed through `Characteristic` now need to come from `api.hap` instead. I need to verify whether the current code uses any of these deprecated patterns and update accordingly. Looking at the constructor signature for `registerAccessory`, it expects a function that receives `log`, `config`, and `api` parameters, but the current implementation is only capturing `log` and `config`. I need to verify what the Homebridge v2 API actually provides to accessory constructors and whether the `api` parameter is

now required or optional. The `homebridge.version` property at line 9 might also need to be updated depending on how the API structure has changed.

Tool: webfetch

Tool: webfetch

Assistant

Thinking:

Now I have a clear picture. Let me analyze what needs to change:

Key findings from the Homebridge v2 source code:

1. Accessory Constructor Signature

From `api.ts`, the `AccessoryPluginConstructor` interface is:

```
interface AccessoryPluginConstructor {  
  new (logger: Logging, config: AccessoryConfig, api: API): AccessoryPlugin;  
}
```

The constructor now receives 3 parameters: `logger`, `config`, and `api`.

The current code at line 17 only accepts `(log, config)`, missing the `api` parameter.

2. HAP Access

The API interface provides:

- `api.hap` - the HAP-NodeJS module (`typeof hapNodeJs`)
- `api.hapLegacyTypes` - for older accessories/platforms
- `api.version` - the homebridge API version as a number (2.7 in v2)
- `api.serverVersion` - the homebridge semver version string

So `homebridge.hap.Service` and `homebridge.hap.Characteristic` still work through `api.hap`.

3. `Characteristic.getValue()` removed

Line 150 of `index.js` uses `state.value` which is correct. The migration guide says `Characteristic.getValue()` has been removed in favor of `Characteristic.value`. The code already uses `.value`, so it's fine.

4. `registerAccessory`

The `registerAccessory` method has two overloads: one that takes just an accessory name and constructor, and another that takes a plugin identifier, accessory name, and constructor. The current code uses the two-argument form, which is still supported.

5. Package dependencies

The `request` package isn't actually used in the code, so I can remove it. `polling-to-event` should stay since it's actively used. Homebridge v2 now requires `Node.js ^22.12.0 || ^24.0.0`.

6. `homebridge.version` VS `api.version`

The code logs `homebridge.version`, which in v2 is `api.version` (2.7, the API version, not the server version). This should still work.

7. The `api` parameter

The constructor receives `api` as the third parameter, so I need to capture it and use `api.hap` to access `Service` and `Characteristic` instead of pulling them from the module export. The `registerAccessory` call stays the same since it's still supported. I should also remove the unused `request` dependency, update the Node.js and homebridge engine requirements, and bump the version number.

Let me double-check the migration guide to see if I'm missing any deprecated patterns like `BatteryService` or `enum` access on `Characteristic`...

Continuing to check the migration guide...

The code doesn't use `getValue()` or `getServiceByUUIDAndSubType()`, so those changes don't apply. For the `api` parameter, I need to capture it in the constructor and use `api.hap` to access `Service` and `Characteristic`, since the old pattern of accessing them directly from the homebridge module won't work in v2. I'm realizing the `homebridge` parameter passed to the initializer is actually the API instance, so I can access `Service` and `Characteristic` through `homebridge.hap`. The key change needed is updating the constructor to accept the `api` parameter as the third argument, since Homebridge v2 expects that signature. Let me verify the HAP-NodeJS v1 exports to make sure `Service` and `Characteristic` are still available through `api.hap`.

Looking at the current code, the `makeHttpGet` function and `polling-to-event` usage should work fine as-is. The main migration steps are updating the constructor signature, removing the unused `request` dependency, and updating the engine constraints in `package.json` to reflect Homebridge v2 requirements.

Now I have a clear picture of the changes needed. Here's the migration plan:

Migration Plan: Homebridge v1.0 → v2.2.1

Breaking Changes Analysis

The key breaking change for this **accessory plugin** (from the HAP-NodeJS v1 / Homebridge v2 migration guide):

1. **Accessory constructor signature changed:** `AccessoryPluginConstructor` now requires `(logger, config, api)` — 3 parameters instead of 2. The `api` parameter is the `HomebridgeAPI` instance.
2. **request package:** Listed in `package.json` but never imported in `index.js` (which uses Node's built-in `http`). Should be removed.
3. **Engine constraints:** Homebridge v2 requires `node: ^22.12.0 || ^24.0.0`.

What Does NOT Need to Change (verified safe)

- `homebridge.registerAccessory` — the 2-arg overload still works
- `homebridge.hap.Service` / `homebridge.hap.Characteristic` — still available via `api.hap`
- `Characteristic.CurrentDoorState` / `TargetDoorState` enums — unchanged

- `Characteristic.value` property — already correct (not the removed `getValue()`)
- `Service.setCharacteristic / getCharacteristic / .on('get', 'set')` — unchanged
- `polling-to-event` package — still compatible
- `Service.GarageDoorOpener / Service.AccessoryInformation` — unchanged

Changes to Make

1. `index.js` — Constructor signature (line 17)

```
- constructor(log, config) {
+ constructor(log, config, api) {
```

Capture `api` and use `api.hap` instead of resolving `Service/Characteristic` from the module-level `homebridge` variable. This is necessary because in v2, the accessory instance is constructed with the `api` directly, and the module-level `homebridge` parameter in the initializer is the API — but the constructor's 3rd param is the authoritative source.

Actually, the current pattern of grabbing `Service/Characteristic` from the `homebridge` passed to the initializer still works (it's the API object with `hap` attached). The critical fix is accepting `api` in the constructor so `homebridge v2` doesn't fail when instantiating the accessory.

2. `index.js` — Log api version (line 9)

Change `homebridge.version` (float API version, e.g. 2.7) to `homebridge.serverVersion` (semver string, e.g. 2.2.1) for more useful logging. Or keep as-is — both exist.

3. `package.json` — Remove unused request dependency

```
"dependencies": {
-   "polling-to-event": "^2.1.0",
-   "request": "^2.83.0"
+   "polling-to-event": "^2.1.0"
}
```

4. `package.json` — Update engine constraints

```
"engines": {
-   "node": ">=8.9.4",
-   "homebridge": ">=0.4.38"
+   "node": "^22.12.0 || ^24.0.0",
+   "homebridge": "^1.6.0 || ^2.0.0"
},
```

5. `package.json` — Bump version

1.1.0 → 2.0.0 (breaking change due to node/homebridge version requirements).

Summary of Edits

File	Change	Reason
index.js:17	Add api param to constructor	HAP-NodeJS v1 requires 3-arg constructor
index.js:9	homebridge.version → api.version (use captured api)	Consistency with v2 API
package.json	Remove request dep	Unused, dead package
package.json	Update engines	v2 requirements
package.json	Bump version to 2.0.0	Breaking change

The functional logic (polling, door state, HTTP requests) remains unchanged.

Shall I proceed with the implementation?

User

yes. proceed.

Assistant

Thinking:

The user confirmed. Let me proceed with the implementation. I'll make all the changes now.

Tool: edit

Tool: edit

Tool: edit

Tool: edit

Tool: edit

Assistant

Thinking:

All edits applied. Let me verify the final state of both files.

Tool: read

Tool: read

Assistant

Thinking:

All changes look correct. Let me verify the syntax is valid by running a quick node check.

Tool: bash

Assistant

Thinking:

Syntax check passed. All changes are complete. Let me summarize.

Done. Here's what changed:

index.js:

- Renamed `homebridge` → `api` parameter for clarity
- Constructor now accepts 3rd `api` parameter (`log`, `config`, `api`) — required by HAP-NodeJS v1
- `api.registerAccessory()` instead of `homebridge.registerAccessory()`
- `api.serverVersion` instead of `homebridge.version`

package.json:

- Version bumped to 2.0.0
 - `engines.node` → `^22.12.0 || ^24.0.0`
 - `engines.homebridge` → `^1.6.0 || ^2.0.0`
 - Removed unused `request` dependency
-