changelog.md for @ecobridge.xyz/devicemanager

2026-01-13 - 3.1.0 - feat(print)

use IPP smartPrint and normalize IPP capabilities and job mapping

- Use IppProtocol.smartPrint for automatic format detection/conversion when submitting print jobs.
- Normalize and map IIppJob -> IPrintJob via mapIppJobToInternal, collapsing extended IPP job states into internal states.
- Parse IIppPrinterCapabilities fields (mediaSizeSupported, mediaTypeSupported, sidesSupported, printQualitySupported, copiesSupported) and derive supportsDuplex from sidesSupported and maxCopies from copiesSupported range with a fallback.
- Map numeric IPP printQuality values (3,4,5) to internal quality strings (draft, normal, high).
- Switched calls to getPrinterAttributes/getJobAttributes and adjusted job listing to map returned IIppJob objects.
- Export new IPP types from protocols index: IIppPrinterCapabilities, IIppJob, IIppPrintOptions.

2026-01-12 - 3.0.2 - fix(devicemanager)

no changes detected - nothing to commit

- git diff indicates no modifications, additions, or deletions
- no files were changed in the provided diff

2026-01-12 - 3.0.1 - fix(release)

add npm registries to release config and expand documentation for UniversalDevice architecture and smart-home features

- npmextra.json: add "registries" to release configuration to publish to both Verdaccio (https://verdaccio.lossless.digital) and the npm registry
- readme.hints.md: rewritten/expanded implementation notes to describe the UniversalDevice architecture, composable features (including smart-home types like light, switch, sensor, climate, cover, lock, fan, camera), protocols, discovery, factories, interfaces, and testing guidance
- readme.md: add factory usage examples including smart-home factory functions, update txtRecords usage (rp) in examples, and small copy/emoji edits

2026-01-10 - 3.0.0 - BREAKING CHANGE(devicemanager)

migrate tests to new UniversalDevice/feature-based API, add device factories, SNMP protocol/feature and IP helper utilities

- Replace protocol-specific device classes (Scanner, Printer) with UniversalDevice and feature objects (ScanFeature, PrintFeature, PlaybackFeature, VolumeFeature, PowerFeature, SnmpFeature)
- Add device factory functions: createScanner, createPrinter, createSpeaker, createUpsDevice
- Add DeviceManager.getDevices selector and updated selectDevice behavior (throws when no match)
- Expose SnmpProtocol and other protocol implementations
- Introduce IP helper utilities: isValidIp, cidrToIps, getLocalSubnet
- Update tests and logging to use feature-based APIs and factories (selectFeature/getFeature, hasFeature, featureCount)

2026-01-09 - 2.3.1 - fix(readme)

update README to comprehensive, TypeScript-first documentation covering installation, quick start, examples, API usage, events, error handling, requirements, credits, and legal/company information

- Rewrote readme.md with ~590 additional lines to provide a full usage guide and examples
- Added installation instructions for pnpm, npm, and yarn and badges for npm and license
- Documented OOP usage pattern (Discovery → Selection → Feature → Operation), event handling, and error handling examples
- Clarified requirements (Node.js 18+, TypeScript 5.0+), credits, license, trademark and company contact information
- Docs-only change — no code or API modifications

2026-01-09 - 2.3.0 - feat(devicemanager)

add selector-based device APIs, selectFeature helper, convenience discovery methods, and ESCL scan completion fallback

- Introduce IDeviceSelector and add selector support to getDevices(selector) to filter devices by id,address,name,model,manufacturer and feature capabilities.
- Add selectDevice(selector) which returns exactly one device (throws if none) and logs a warning if multiple matches are returned without a unique identifier.
- Deprecate getDevice(id) and getDeviceByAddress(address) in favor of selector-based retrieval methods.
- Add private matchesSelector(...) implementing exact identity checks (id,address), case-insensitive partial attribute matching (name, model, manufacturer), and feature availability checks (hasFeature, hasFeatures, hasAnyFeature).
- Add selectFeature(type) on devices to provide fail-fast access to a required feature (throws if missing).
- Add discoverScanners(subnet, options) and discoverPrinters(subnet, options) convenience methods that run targeted network scans and return discovered scanners or printers respectively.
- Improve ESCL protocol waitForScanComplete to attempt a direct download first (which triggers/blocks on many scanners) and fall back to polling if direct download fails or returns empty data.

2026-01-09 - 2.2.0 - feat(smarthome)

add smart home features and Home Assistant integration (WebSocket protocol, discovery, factories, interfaces)

- Add concrete smart home feature implementations: light, climate, sensor, switch, cover, lock, fan, camera.
- Introduce Home Assistant WebSocket protocol handler (protocol.homeassistant) and Home Assistant discovery via mDNS (discovery.classes.homeassistant).
- Add generic smart home interfaces and Home Assistant-specific interfaces (smarthome.interfaces, homeassistant.interfaces) and export them.
- Add smart home factories to create devices for discovered/declared smart home entities and export factory helpers.
- Update plugins to include WebSocket (ws) and add ws dependency and @types/ws in package.json.

2026-01-09 - 2.1.0 - feat(devicemanager)

prefer higher-priority discovery source when resolving device names and track per-device name source

- Add TNameSource type and NAME_SOURCE_PRIORITY to rank name sources (generic, manual, airplay, chromecast, mdns, dlna, sonos).
- Replace chooseBestName with shouldUpdateName that validates 'real' names and uses source priority when deciding to update a device name.
- Add nameSourceByIp map to track which discovery source provided the current name and persist updates during registration.
- Register devices with an explicit nameSource (e.g. 'mdns', 'dlna', 'sonos', 'manual') and map speaker protocols to appropriate name sources.
- Ensure manual additions use 'manual' source and non-real names default to 'generic'.
- Clear nameSourceByIp entries when devices are removed/disconnected and on shutdown.

2026-01-09 - 2.0.0 - BREAKING CHANGE(core)

rework core device architecture: consolidate protocols into a protocols/ module, introduce UniversalDevice + factories, and remove many legacy device-specific classes (breaking API changes)

- Consolidated protocol implementations into ts/protocols and added protocols/index.ts for unified exports.

- Added device factory layer at ts/factories/index.ts to create UniversalDevice instances with appropriate features.
- Introduced protocols/protocol.upssnmp.ts (UPS SNMP handler) and other protocol reorganizations.
- Removed legacy concrete device classes and related files (Device abstract, Scanner, Printer, SnmpDevice, UpsDevice, DlnaRenderer/Server, Speaker and Sonos/AirPlay/Chromecast implementations).
- Updated top-level ts/index.ts exports to prefer UniversalDevice, factories and the new protocols module.
- Updated feature and discovery modules to import protocols from the new protocols index (import path changes).
- BREAKING: Consumers must update imports and device creation flows to use the new factories/UniversalDevice and protocols exports instead of the removed legacy classes.

2026-01-09 - 1.1.0 - feat(devicemanager)

Introduce a UniversalDevice architecture with composable Feature system; add extensive new device/protocol support and discovery/refactors

- Add UniversalDevice class and Feature abstraction with concrete features: scan, print, playback, volume, power, snmp, dlna-render/serve.
- Add SSDP discovery and DLNA implementations (renderer + server) and integrate SSDP into DeviceManager.
- Add speaker subsystem and concrete speaker implementations: Sonos, AirPlay, Chromecast, plus generic Speaker API and Volume/Playback features.
- Add SNMP feature and SNMP device handling plus UPS support (NUT and UPS SNMP handlers and UpsDevice).
- Refactor protocol implementations: move/replace scanner/printer protocol code into protocols/ (eSCL, SANE, IPP) and update network scanner to probe additional ports (AirPlay, Sonos, Chromecast) and device types.
- Update exports (ts/index.ts) to expose new modules, types and helpers; update plugins import handling (node-ssdp default export compatibility).
- Add developer docs readme.hints.md describing new architecture and feature APIs, and various helper fixes (iprange/os import, typed socket handlers).

2026-01-09 - 1.0.1 - initial

Initial project release.

- Project initialized (initial commit).

- Duplicate initial commits consolidated into this release.

Revision #2
Created 2026-03-28 13:06:50 UTC by foss.global Team
Updated 2026-03-29 16:49:16 UTC by foss.global Team