

changelog.md for @git.zone/cli

2026-03-24 - 2.13.15 - fix(repo)

no changes to commit

2026-03-24 - 2.13.14 -
fix(mod_format)

move smartconfig file renaming into the formatter orchestrator

- Renames smartconfig.json or npmextra.json to .smartconfig.json before formatters run
- Simplifies the smartconfig formatter to only read and modify .smartconfig.json
- Removes create/delete change planning for config renames and applies only content updates within the formatter

2026-03-24 - 2.13.13 - fix(vscode-
template)

update VS Code schema matching to use .smartconfig.json

- Changes the VS Code template settings so the JSON schema applies to /.smartconfig.json instead of /npmextra.json.

2026-03-24 - 2.13.12 - fix(mod_format)

render format templates through smartscf before comparing generated files

- adds smartscf-based in-memory template rendering so supplied variables are applied before detecting changes
- supports release.accessLevel as a fallback when selecting public vs private CI templates
- matches rendered output by template or destination path to handle renamed files from template frontmatter

2026-03-24 - 2.13.10 - fix(config)

migrate configuration handling from npmextra to smartconfig

- replace @push.rocks/npmextra with @push.rocks/smartconfig across config, commit, format, and service modules
- switch managed project config file references from npmextra.json to smartconfig.json
- update formatting and package metadata checks to include smartconfig.json
- extend the gitignore template with Rust build output directories

2026-03-11 - 2.13.9 - fix(deps,readme)

bump dependencies and update README to prefer pnpm and document semantic commit flags

- Dev dependency updates: @git.zone/tsbuild -> ^4.3.0, @git.zone/tstest -> ^3.3.2, @types/node -> ^25.4.0
- Dependency upgrades: @git.zone/tsdoc -> ^2.0.0, @git.zone/tspublish -> ^1.11.2, @push.rocks/lik -> ^6.3.1, @push.rocks/smartfs -> ^1.5.0, @push.rocks/smartlog -> ^3.2.1, @push.rocks/smartstream -> ^3.4.0, prettier -> ^3.8.1 (and other minor/patch bumps)
- README changes: prefer pnpm for global install, clarify format command dry-run behavior and --write flag, add and document gitzone commit flags (-y/--yes, -p/--push, -t/--test, -b/--build, -r/--release) and AI-powered commit workflow

- No source code changes; this is a documentation and dependency refresh, recommend a patch release

2026-03-05 - 2.13.8 - fix(dependencies)

move runtime tooling packages from devDependencies to dependencies

- Removed @push.rocks/smartdelay, @push.rocks/smartinteract, @push.rocks/smartnetwork, and @push.rocks/smartshell from devDependencies and added them to dependencies
- No package version numbers were changed; this ensures the moved packages are installed for consumers at runtime

2026-03-05 - 2.13.7 - fix(deps)

bump devDependencies: @git.zone/tsbuild to ^4.1.4 and @push.rocks/smartshell to ^3.3.7

- Updated @git.zone/tsbuild from ^4.1.2 to ^4.1.4 (patch)
- Updated @push.rocks/smartshell from ^3.3.0 to ^3.3.7 (patch)

2026-02-01 - 2.13.6 - fix(templates/npm)

use tsbuild tsfolders instead of --web flag in npm template build script

- Changed build script in assets/templates/npm/.package.json from "(tsbuild --web --allowimplicitany)" to "(tsbuild tsfolders --allowimplicitany)"
- Replaces --web flag with explicit tsfolders argument to correctly target project folders during build

2026-02-01 - 2.13.5 - fix(templates/npm)

update npm template: tweak test script, bump devDependencies, add smartpath dependency, and fix ts import path

- test script updated: '(tstest test/ --web)' -> '(tstest test/ --verbose --logfile --timeout 60)'
- devDependencies bumped: @git.zone/tsbuild ^3.1.2 -> ^4.1.2, @git.zone/tsrun ^2.0.0 -> ^2.0.1, @git.zone/tstest ^3.1.3 -> ^3.1.8, @types/node ^24.10.1 -> ^25.2.0
- dependencies: added @push.rocks/smartpath ^6.0.0
- TypeScript template import fixed: './{module.name}.plugins.js' -> './plugins.js'

2025-12-18 - 2.13.3 - fix(tsconfig)

remove experimentalDecorators and useDefineForClassFields from TypeScript configuration files

- Removed "experimentalDecorators": true from assets/templates/multienv/deno.json and tsconfig.json
- Removed "useDefineForClassFields": false from tsconfig.json
- This change alters TypeScript/Deno compiler behavior: decorator support and legacy class-field initialization semantics may be affected; code relying on those may need updates

2025-12-16 - 2.13.2 - fix(deps)

bump @git.zone/tspublish to ^1.11.0

- Updated dependency @git.zone/tspublish from ^1.10.3 to ^1.11.0 in package.json

2025-12-16 - 2.13.1 - fix(npmextra)

merge old npmextra keys into new keys during migration, preserving existing new values

- Changed migration logic to merge data when both old and new keys exist instead of skipping the merge.
- Merge preserves existing new-key values (old values do not overwrite new ones) and still deletes the old key after migration.
- Applied the fix in both `ts/mod_format/format.npmextra.ts` and `ts/mod_format/formatters/npmextra.formatter.ts`.
- Adds a console log for successful migrations; behavior for single-key rename remains unchanged.

2025-12-16 - 2.13.0 - feat(tests)

feat(tests): add sandbox test fixture, CI and editor configs; bump deps

- Added comprehensive test/ fixture (sandbox-npmts) including `package.json`, `npmextra.json`, `readme`, `GitLab CI (.gitlab-ci.yml)`, `.npmrc`, `VSCoDe launch/settings` and `qenv.yml`
- Added test sources and helper files under `test/test` and `test/ts` (browser and node tests, commitinfo data, simple library code)
- Updated dependencies in `package.json`: `@git.zone/tsdoc -> ^1.11.4`, `@push.rocks/smartfs -> ^1.3.1`

2025-12-15 - 2.12.2 - fix(cli)

noop: no changes

- No source or documentation changes detected in the diff; nothing to release.
- Package version remains unchanged at 2.12.1.

2025-12-15 - 2.12.1 - fix(cli)

No changes detected — no version bump required

- Current package version: 2.12.0
- No files changed in this commit
- No release or version bump necessary

2025-12-15 - 2.12.0 - feat(ci,test)

feat(ci/test): add test scaffold, GitLab CI, update gitea workflows and .gitignore

- Add comprehensive test/ scaffold including sample tests, test package.json, npmextra.json, test fixtures and TypeScript test sources
- Add GitLab CI pipeline (test/.gitlab-ci.yml), test-specific .npmrc, VSCode launch/settings and supporting qenv/readme files for CI/local test runs
- Update .gitea workflow YAML files to use templated placeholders and corrected container image path for CI execution
- Update .gitignore to exclude AI assistant and tooling dirs (.claude/, .serena/) and add test/.gitignore to ignore test artifacts
- Update changelog and documentation files to reflect recent formatter/commit/service changes and the new test/CI additions

2025-12-15 - 2.11.1 -

fix(mod_format/formatters)

fix(packagejson.formatter): correctly parse scoped package dependency arguments and default to latest

- Handle scoped packages (e.g. @scope/name@version) by detecting the last '@' after the scope slash so package name and version are split correctly.
- Fallback to 'latest' when no version is provided.
- Fixes earlier incorrect splitting on every '@' which broke scoped package names.

2025-12-15 - 2.11.0 -

feat(mod_format)

feat(mod_format): use unified diff formatter with filenames and context in BaseFormatter.displayDiff

- Replaced plugins.smartdiff.formatLineDiffForConsole(...) with plugins.smartdiff.formatUnifiedDiffForConsole(...) when both before and after are present.
- Passes originalFileName and revisedFileName as diff.path and sets context to 3 to show a unified diff with surrounding lines.
- Improves console output for multi-line diffs by using unified diff format and including file names.

2025-12-15 - 2.10.0 - feat(mod_format)

Refactor formatting modules to new BaseFormatter and implement concrete analyze/apply logic

- Replace generic LegacyFormatter with explicit BaseFormatter implementations for formatters: copy, gitignore, license, npmextra, packagejson, prettier, readme, templates, tsconfig (legacy.formatter.ts removed).
- Copy formatter: implemented pattern-based copying, template-preserve path handling, content equality check and planned change generation/apply.
- Gitignore formatter: canonical template with preservation of custom section when updating/creating .gitignore.
- License formatter: added runtime license check against node_modules for incompatible licenses and reporting (no file changes).
- Npmextra formatter: automatic migrations for old namespace keys to package-scoped keys and migration of npmAccessLevel -> @git.zone/cli.release.accessLevel; reformatting and interactive prompting to fill missing repo metadata.
- Package.json formatter: enforces repository/metadata, sets module type/private/license/scripts/files, ensures/updates dependencies (including fetching latest via registry), and applies pnpm overrides from assets.
- Prettier formatter: added check() to compute diffs by running Prettier and returning per-file before/after diffs.
- Readme formatter: create readme.md and readme.hints.md when missing with default content.
- Templates formatter: apply templates from templatesDir based on project type (vscode, CI, docker, website/service/wcc), compare template vs destination and create/modify files as needed; ensures dest directories exist.
- Tsconfig formatter: sets compilerOptions.baseUrl and computes path mappings from @git.zone/tspublish modules.
- General: extensive use of plugins (smartfs, path, smartnpm, smartinteract, smartobject, smartlegal), improved logging and verbose messages.

2025-12-15 - 2.9.0 - feat(format)

Add --diff option to format command to display file diffs; pass flag through CLI and show formatter diffs. Bump @git.zone/tsdoc to ^1.11.0.

- Add a diff boolean option to mod_format to enable showing file diffs during format operations.

- CLI change: pass `argvArg.diff` into the options so the `--diff` flag is honored by the format command.
- When diff is enabled, run `formatter.check()` for each active formatter and call `displayAllDiffs()` for those with differences, with informational logging.
- Update dependency `@git.zone/tsdoc` from `^1.10.2` to `^1.11.0`.

2025-12-15 - 2.8.0 - feat(commit)

Add commit configuration and automatic pre-commit tests

- Add `CommitConfig` class to manage `@git.zone/cli.commit` settings in `npmextra.json` (`alwaysTest`, `alwaysBuild`).
- Export `CommitConfig` from `mod_config` for use by the CLI.
- Add 'gitzone config commit' subcommand with interactive and direct-setting modes (`alwaysTest`, `alwaysBuild`).
- Merge CLI flags and `npmextra` config: `-t/--test` and `-b/--build` now respect `commit.alwaysTest` and `commit.alwaysBuild`.
- Run 'pnpm test' early in the commit flow when tests are enabled; abort the commit on failing tests and log results.
- Update commit UI/plan to show the test option and include the test step when enabled.
- Add 'gitzone config services' entry to configure services via `ServiceManager`.

2025-12-14 - 2.7.0 - feat(mod_format)

Add check-only formatting with interactive diff preview; make formatting default to dry-run and extend formatting API

- Add `BaseFormatter.check()`, `displayDiff()` and `displayAllDiffs()` to compute and render diffs without applying changes.
- Extend `runFormatter` API with new options: `write` (use to apply changes), `checkOnly` (only check for diffs), and `showDiff` (display diffs). When `checkOnly` is used, `runFormatter` returns an `ICheckResult`.
- Change default formatting behavior to dry-run. Use `--write / -w` to actually apply changes. CLI format command updated to respect `--write/-w`.
- Add `formatNpmextraWithDiff` in `mod_config` to preview diffs for `npmextra.json` and prompt the user before applying changes; calls to `add/remove/clear` registries and `set access level` now use this preview flow.

- `Project.fromCwd` now accepts an options object (`{ requireProjectType?: boolean }`) so callers can skip the `projectType` requirement when appropriate; `runFormatter` no longer requires `projectType` for certain formatters.
- Introduce a list of formatters that don't require `projectType`: `npmextra`, `prettier`, `cleanup`, `packagejson`.
- Export the `ICheckResult` type from the formatter module and update `mod_format` interfaces to include `ICheckResult`.
- Bump dependency `@push.rocks/smartdiff` to `^1.1.0`.

2025-12-14 - 2.6.1 - fix(npmextra)

Normalize `npmextra.json`: move `tsdoc` legal entry and reposition `@git.zone/cli` configuration

- Move `TSDoc` legal text into a top-level `"tsdoc.legal"` property in `npmextra.json`
- Remove the misplaced `"gitzone"` object that previously contained the `TSDoc` legal block
- Reorder and properly nest the `"@git.zone/cli"` configuration to follow the new `"tsdoc"` section
- Pure configuration change (JSON structure) — no functional code changes

2025-12-14 - 2.6.0 - feat(mod_commit)

Add execution plan output to commit command

- Print an execution plan at the start of the commit flow (shows active options and planned steps)
- New `printExecutionPlan(options)` added to `ts/mod_commit/mod.ui.ts`
- Commit command now calls `ui.printExecutionPlan(...)` before running `format` or other steps
- Execution plan reflects flags: `auto-accept (-y)`, `push (-p)`, `build (-b)`, `release (-r)`, `--format`, and target registries

2025-12-14 - 2.5.0 - feat(mod_standard)

Add interactive main menu and help to standard CLI module; route commands via dynamic imports

- Introduce interactive CLI menu using @push.rocks/smartinteract to prompt user for actions.
- Add routing to commands (commit, format, config, template, services, open, help) using dynamic imports.
- Add showHelp() helper to display usage and available commands.
- Remove previous static template listing and logger.warn placeholder.

2025-12-14 - 2.4.0 - feat(cli)

Add optional build step to release flow and auto-format npmextra config when registries change

- Introduce a --build/-b flag in the commit/release flow to run 'pnpm build' before pushing/releases
- Verify the working tree is clean after the build and abort the release if build produces uncommitted changes
- Increase total step counting to include build and verification steps in the UI progress output
- Add a runFormatter utility to the formatting module to execute a single formatter programmatically
- Wire runFormatter('npmextra') into mod_config so npmextra.json is formatted automatically after add/remove/clear/access operations
- Add npmextra registry config entry (<https://verdaccio.lossless.digital>) to npmextra.json

2025-12-14 - 2.3.0 - feat(config)

Add interactive menu and help to config command, handle unknown commands, and bump dependencies

- When running the 'config' command with no arguments, show an interactive menu (via SmartInteract) to choose actions (show, add, remove, clear, access, help) instead of defaulting to 'show'.
- Add explicit 'help' subcommand and log an error for unknown commands before showing help.
- Update devDependencies: @git.zone/tsbuild -> ^4.0.2, @types/node -> ^25.0.2.
- Update dependency: @push.rocks/smartjson -> ^6.0.0.

2025-12-04 - 2.2.1 - fix(commit)

Prevent auto-accept for BREAKING CHANGE commits; require manual confirmation and warn when --yes is used

- Do not auto-accept AI commit recommendations when the suggested change is a BREAKING CHANGE (major bump).
- Only auto-accept recommendations when -y/--yes is provided and the change is not breaking.
- When --yes is provided but the recommendation is a breaking change, log a warning and fall back to interactive confirmation.
- Introduced isBreakingChange and canAutoAccept flags to centralize the auto-accept logic.

2025-12-02 - 2.2.0 - feat(services)

Improve services manager and configuration; switch test templates to @git.zone/tstest; bump dev dependencies and update docs

- services: Add robust ServiceConfiguration (creates .nogit/env.json with sane defaults, syncs ports from existing Docker containers, validates and can reconfigure ports)
- services CLI: improved start/stop/restart flows, better logging/help output and enhanced global commands (list/status/stop/cleanup)
- templates/tests: replace @push.rocks/tapbundle with @git.zone/tstest and update template test.ts to export default tap.start()
- format: stop auto-updating tslint template and mark @push.rocks/tapbundle as deprecated in package formatting logic
- dependencies: bump @git.zone/tsbuild, @git.zone/tsrun, @git.zone/tstest, @git.zone/tsdoc, @push.rocks/projectinfo, @push.rocks/smarts, @push.rocks/smartsfs, prettier and other dev deps
- docs: README updates — add issue reporting/security section, AI-powered commit recommendation notes, and clarify trademark/legal wording

2025-11-29 - 2.1.0 - feat(mod_services)

Add global service registry and global commands for managing project containers

- Introduce GlobalRegistry class to track registered projects, their containers, ports and last activity (ts/mod_services/classes/globalregistry.ts)
- Add global CLI mode for services (use -g/--global) with commands: list, status, stop, cleanup (ts/mod_services/index.ts)

- ServiceManager now registers the current project with the global registry when starting services and unregisters when all containers are removed (ts/mod_services/classes.servicemanager.ts)
- Global handlers to list projects, show aggregated status, stop containers across projects and cleanup stale entries
- Bump dependency @push.rocks/smartfile to ^13.1.0 in package.json

2025-11-27 - 2.0.0 - BREAKING CHANGE(core)

Migrate filesystem to smartfs (async) and add Elasticsearch service support; refactor format/commit/meta modules

- Replace @push.rocks/smartfile usage with @push.rocks/smartfs across the codebase; all filesystem operations are now async (SmartFs.file(...).read()/write(), SmartFs.directory(...).list()/create()/delete(), etc.)
- Convert formerly synchronous helpers and APIs to async (notable: detectProjectType, getProjectName, readCurrentVersion and related version bumping logic). Callers updated accordingly.
- Add Elasticsearch support to services: new config fields (ELASTICSEARCH_*), Docker run/start/stop/logs/status handling, and ELASTICSEARCH_URL in service configuration.
- Refactor formatting subsystem: cache and rollback/backup systems removed/disabled for stability, format planner execution simplified (sequential), diff/stats reporting updated to use smartfs.
- Update package.json dependencies: bump @git.zone/tsbuild, tsrun, tstest; upgrade @push.rocks/smartfile to v13 and add @push.rocks/smartfs dependency; update @types/node.
- Update commit flow and changelog generation to use smartfs for reading/writing files and to await version/branch detection where necessary.
- Expose a SmartFs instance via plugins and adjust all mod.* plugin files to import/use smartfs where required.
- Breaking change: Public and internal APIs that previously used synchronous smartfile APIs are now asynchronous. Consumers and scripts must await these functions and use the new smartfs API.

2025-11-17 - 1.21.5 - fix(tsconfig)

Remove emitDecoratorMetadata from tsconfig template

- Removed the "emitDecoratorMetadata" compiler option from assets/templates/tsconfig_update/tsconfig.json
- This updates the tsconfig template to avoid emitting decorator metadata when targeting ES2022

2025-11-17 - 1.21.4 - fix(tsconfig template)

Remove experimentalDecorators and useDefineForClassFields from tsconfig template

- Removed experimentalDecorators option from assets/templates/tsconfig_update/tsconfig.json
- Removed useDefineForClassFields option from assets/templates/tsconfig_update/tsconfig.json

2025-11-17 - 1.21.3 - fix(assets/templates/multienv)

Remove unused Bun configuration template (assets/templates/multienv/bunfig.toml)

- Deleted assets/templates/multienv/bunfig.toml which previously provided Bun TypeScript decorator configuration
- Cleans up stale/unused template to avoid shipping obsolete Bun config
- No functional code changes; removes an unused asset file

2025-11-17 - 1.21.2 - fix(templates/multienv)

Disable useDefineForClassFields in multienv TypeScript configs to ensure decorator compatibility

- Set useDefineForClassFields = false in assets/templates/multienv/bunfig.toml to keep Bun's transpiler compatible with decorator usage

- Set "useDefineForClassFields": false in assets/templates/multienv/deno.json to ensure Deno/TypeScript compiler emits class fields compatible with decorators

2025-11-17 - 1.21.1 - fix(templates.multienv)

Enable checkJs in multienv Deno template to enable JS type checking

- Added "checkJs": true to compilerOptions in assets/templates/multienv/deno.json to enable JavaScript type checking for the Deno multienv template

2025-11-17 - 1.21.0 - feat(multienv)

Add multi-env templates enabling TypeScript decorators for Bun and Deno; rename npmextra config key to szci

- Added assets/templates/multienv/bunfig.toml to enable Bun TypeScript transpiler experimentalDecorators
- Added assets/templates/multienv/deno.json with experimentalDecorators, lib and target set for ES2022
- Updated npmextra.json: renamed top-level config key from "npmci" to "szci" (keeps npmGlobalTools, npmAccessLevel and npmRegistryUrl unchanged)

2025-11-06 - 1.20.0 - feat(commit)

Add non-interactive --yes (-y) flag to commit command to auto-accept AI recommendations and optionally push with -p

- Add -y / --yes flag to gitzone commit to auto-accept AI-generated commit recommendations without interactive prompts
- Support -yp or -y -p combinations to auto-accept and push to origin; -p / --push remains the separate control for pushing
- Implementation creates a smartinteract AnswerBucket programmatically when -y is used and populates commitType, commitScope, commitDescription and pushToOrigin

- Preserves existing UI output and interactive flow when -y is not used; fully backward compatible and CI/CD friendly
- Updated CLI usage and documentation (readme.hints.md) to document the new flags

2025-11-05 - 1.19.9 - fix(mod_commit)

Refactor version bumping to a unified implementation for npm and Deno; remove npm-exec based helpers and add file-based version readers/updaters to avoid npm warning pollution

- Removed legacy npm/deno-specific helpers (bumpNpmVersion, syncVersionToDenoJson, bumpDenoVersion) that relied on executing npm and caused warning pollution
- Added readCurrentVersion() to read version from package.json or deno.json
- Added updateVersionFile() helper to write version directly into JSON files
- Added unified bumpProjectVersion() that handles npm, deno and both with a single code path; reuses calculateNewVersion()
- Stages updated files, commits v<newVersion> and creates a tag v<newVersion>
- Benefits: no npm warning pollution in deno.json, simpler git history, consistent behavior across project types

2025-11-04 - 1.19.8 - fix(package.json)

Bump @git.zone/tsdoc dependency to ^1.9.2

- Updated dependency @git.zone/tsdoc from ^1.9.1 to ^1.9.2 in package.json

2025-11-04 - 1.19.7 - fix(dependencies)

Bump @git.zone/tsdoc to ^1.9.1

- Updated package.json dependency @git.zone/tsdoc from ^1.9.0 to ^1.9.1

2025-11-04 - 1.19.6 - fix(cli)

Bump @git.zone/tsdoc dependency to ^1.9.0

- Updated dependency @git.zone/tsdoc from ^1.8.3 to ^1.9.0 in package.json

2025-11-04 - 1.19.5 - fix(cli)

Bump @git.zone/tsdoc to ^1.8.3 and add local .claude settings for allowed permissions

- Updated dependency @git.zone/tsdoc from ^1.8.2 to ^1.8.3
- Added .claude/settings.local.json to declare allowed permissions for local tooling (Bash commands, Docker, npm, WebFetch and MCP actions)

2025-11-03 - 1.19.3 - fix(tsdoc)

Bump @git.zone/tsdoc to ^1.8.0 and add .claude local settings

- Upgrade dependency @git.zone/tsdoc from ^1.6.1 to ^1.8.0 in package.json
- Add .claude/settings.local.json for local assistant permissions/configuration

2025-11-03 - 1.19.2 - fix(tsdoc)

Bump @git.zone/tsdoc to ^1.6.1 and add .claude/settings.local.json

- Update dependency @git.zone/tsdoc from ^1.6.0 to ^1.6.1
- Add .claude/settings.local.json to include local Claude settings/permissions

2025-11-02 - 1.19.1 - fix(dependencies)

Bump dependencies and add local Claude settings

- Bump devDependencies: @git.zone/tsbuild -> ^2.7.1, @git.zone/tsrun -> ^1.6.2, @git.zone/tstest -> ^2.7.0
- Upgrade runtime dependencies: @git.zone/tsdoc -> ^1.6.0; update @push.rocks packages (smartcli ^4.0.19, smartjson ^5.2.0, smartlog ^3.1.10, smartnetwork ^4.4.0, etc.)
- Add .claude/settings.local.json (local project permissions/settings file)

2025-10-23 - 1.19.0 - feat(mod_commit)

Add CLI UI helpers and improve commit workflow with progress, recommendations and summary

- Introduce ts/mod_commit/mod.ui.ts: reusable CLI UI helpers (pretty headers, sections, AI recommendation box, step printer, commit summary and helpers for consistent messaging).
- Refactor ts/mod_commit/index.ts: use new UI functions to display AI recommendations, show step-by-step progress for baking commit info, generating changelog, staging, committing, bumping version and optional push; include commit SHA in final summary.
- Enhance ts/mod_commit/mod.helpers.ts: bumpProjectVersion now accepts currentStep/totalSteps to report progress and returns a consistent newVersion after handling npm/deno/both cases.
- Add .claude/settings.local.json: local permissions configuration for development tooling.

2025-10-23 - 1.18.9 - fix(mod_commit)

Stage and commit deno.json when bumping/syncing versions and create/update git tags

- bumpDenoVersion now creates a Smartshell instance and runs git add deno.json, git commit -m "v<newVersion>", and git tag v<newVersion> to persist the version bump
- syncVersionToDenoJson now stages deno.json, amends the npm version commit with --no-edit, and recreates the tag with -fa to keep package.json and deno.json in sync
- Added informative logger messages after creating commits and tags

2025-10-23 - 1.18.8 - fix(mod_commit)

Improve commit workflow: detect project type and current branch; add robust version bump helpers for npm/deno

- Add mod_commit/mod.helpers.ts with utilities: detectCurrentBranch(), detectProjectType(), bumpProjectVersion(), bumpDenoVersion(), bumpNpmVersion(), syncVersionToDenoJson(), and calculateNewVersion()
- Refactor ts/mod_commit/index.ts to use the new helpers: bumpProjectVersion(projectType, ...) instead of a hard npm version call and push the actual current branch instead of hardcoding 'master'
- Support bumping versions for npm-only, deno-only, and hybrid (both) projects and synchronize versions from package.json to deno.json when applicable
- Improve branch detection with a fallback to 'master' and informative logging on detection failures
- Add local Claude settings file (.claude/settings.local.json) (editor/CI config) — no code behavior change but included in diff

2025-09-07 - 1.18.7 - fix(claude)

Add .claude local settings to whitelist dev tool permissions

- Add .claude/settings.local.json to configure allowed permissions for local AI/tooling helpers (Bash commands, WebFetch, and mcp_serena actions).
- Disable enableAllProjectMcpServers (set to false) to limit automatic project MCP server usage.

2025-09-07 - 1.18.6 - fix(deps)

Bump dependency versions and add local Claude settings

- Updated devDependencies: @git.zone/tsbuild ^2.6.4 → ^2.6.8, @git.zone/tstest ^2.3.4 → ^2.3.6, @push.rocks/smartfile ^11.2.5 → ^11.2.7
- Updated dependencies: @git.zone/tsdoc ^1.5.1 → ^1.5.2, @git.zone/tspublish ^1.10.1 → ^1.10.3, @push.rocks/smartlog ^3.1.8 → ^3.1.9, @push.rocks/smarnpm ^2.0.4 → ^2.0.6, @push.rocks/smartscaf ^4.0.17 → ^4.0.19

- Added `.claude/settings.local.json` to configure local Claude permissions/settings

2025-08-17 - 1.18.5 - fix(dependencies)

Bump smartshell and smartscaf versions; add `.claude` local settings

- Update `@push.rocks/smartshell` from `^3.2.4` to `^3.3.0` in `package.json`
- Update `@push.rocks/smartscaf` from `^4.0.16` to `^4.0.17` in `package.json`
- Add `.claude/settings.local.json` for local assistant permissions/configuration

2025-08-17 - 1.18.4 - fix(cli)

Update dependencies, add local Claude settings, and update gitignore template

- Bump several dependencies: `@git.zone/tsbuild` -> `^2.6.4`, `@git.zone/tspublish` -> `^1.10.1`, `@git.zone/tstest` -> `^2.3.4`, `@push.rocks/smartfile` -> `^11.2.5`, `@push.rocks/npmextra` -> `^5.3.3`, `@push.rocks/smartchok` -> `^1.1.1`, `@push.rocks/smartlog` -> `^3.1.8`, `@push.rocks/smartpath` -> `^6.0.0`, `prettier` -> `^3.6.2`
- Add `.claude/settings.local.json` with local permissions configuration for AI tooling
- Update `assets/templates/gitignore` to ignore `.claude/` and `.serena/` directories
- Add `pnpm onlyBuiltDependencies` entries: `esbuild` and `mongodb-memory-server`

2025-08-16 - 1.18.3 - fix(services)

Simplify S3 endpoint handling in `ServiceConfiguration` to store host only

- `S3_ENDPOINT` now stores the raw host (e.g. `'localhost'`) instead of a full URL with protocol and port.
- Default `.nogit/env.json` creation uses the host-only `S3_ENDPOINT`.
- Sync/update logic (when syncing with Docker or reconfiguring ports) sets `S3_ENDPOINT` to the host only.
- Consumers that previously relied on `S3_ENDPOINT` containing protocol and port should now construct the full endpoint URL using `S3_USESSL`, `S3_HOST` and `S3_PORT`.

2025-08-16 - 1.18.1 - fix(services)

Improve services and commit flow: stop AiDoc, use silent docker inspect, sync ports with logging, fix config loading, and bump deps

- Ensure AiDoc is stopped after building commit recommendation to avoid resource leaks
- Use execSilent for `docker inspect` in DockerContainer to avoid shell noise and improve JSON parsing
- Sync Docker-exposed ports into service configuration with explicit notes (logs) when MongoDB / S3 ports are updated
- Fix synchronous config loading by removing an unnecessary await in ServiceConfiguration.loadConfig
- Bump dependencies: @push.rocks/smartshell -> ^3.2.4, @git.zone/tsdoc -> ^1.5.1
- Add pnpm.onlyBuiltDependencies for puppeteer and sharp to package.json
- Add local Claude settings file (.claude/settings.local.json) with development permissions

2025-08-16 - 1.18.0 - feat(services)

Add Docker port mapping sync and reconfigure workflow for local services

- Add getPortMappings to DockerContainer to extract port bindings from docker inspect output
- Sync existing container port mappings into .nokit/env.json when loading/creating service configuration
- Validate and automatically update ports only when containers are not present; preserve container ports when containers exist
- Recreate containers automatically if detected container port mappings differ from configuration (MongoDB and MinIO)
- Add reconfigure method and new CLI command to reassign ports and optionally restart services
- Improve status output to show configured ports and port availability information
- Minor helpers and imports updated (DockerContainer injected into ServiceConfiguration)
- Add .claude/settings.local.json (local permissions config) to repository

2025-08-15 - 1.17.5 - fix(services)

Update S3 credentials naming and add S3_ENDPOINT/S3_USESSL support for improved MinIO integration

- Replaced S3_USER/S3_PASS with S3_ACCESSKEY/S3_SECRETKEY in ServiceConfiguration
- Added S3_ENDPOINT field with automatic protocol selection based on S3_USESSL
- Introduced S3_USESSL boolean field for SSL/TLS configuration
- Updated ServiceManager logging to display new S3_USESSL configuration
- Added .claude/settings.local.json for local permission settings

2025-08-15 - 1.17.4 - fix(services)

Update S3 credentials naming and add S3_ENDPOINT/S3_USESSL support for improved MinIO integration

- Replaced S3_USER/S3_PASS with S3_ACCESSKEY/S3_SECRETKEY in ServiceConfiguration
- Added S3_ENDPOINT field with automatic protocol selection based on S3_USESSL
- Added S3_USESSL boolean field for SSL/TLS configuration support
- Updated ServiceManager to use new credential names in container setup and logging

2025-08-15 - 1.17.3 - fix(serviceconfig)

Update service configuration to include dynamic MongoDB connection string and add local permissions settings

- Added .claude/settings.local.json for local permissions configuration
- Updated ServiceConfiguration to compute and update MONGODB_URL based on current config values

2025-08-15 - 1.17.2 - fix(ci-test-services)

Update CI/CD configurations, test settings, and Docker service for MongoDB.

- Add .claude/settings.local.json with updated permission settings

- Introduce new GitLab CI, VSCode launch and settings, and updated test configuration files (.gitignore, .npmrc, npmextra.json, package.json, qenv.yml, readme.md)
- Update test scripts in test/test and test/ts to improve project validation
- Fix MongoDB Docker container command by adding '--bind_ip_all' for proper network binding

2025-08-15 - 1.17.1 - fix(services)

Improve services module logging and enhance MongoDB Compass integration

- Refactored services module to use centralized logger from gitzone.logging.ts
- Automatically display MongoDB Compass connection string when starting services or checking status
- Removed custom printMessage wrapper in favor of standard logger.log() calls
- Consistent logging across all service commands

2025-08-14 - 1.17.0 - feat(services)

Add comprehensive development services management for MongoDB and MinIO containers

- Implemented `gitzone services` command for managing local development services
- Added MongoDB and MinIO (S3-compatible) container orchestration
- Smart port assignment (20000-30000 range) to avoid conflicts between projects
- Project-specific container names for complete isolation
- Data persistence in `.nogit/` directories
- MongoDB Compass connection string generation with network IP detection
- Auto-configuration via `.nogit/env.json` with secure defaults
- Commands: start, stop, restart, status, config, compass, logs, remove, clean
- Interactive confirmations for destructive operations

2025-08-08 - 1.16.10 - fix(format)

Improve concurrency control in caching and rollback modules, refine gitignore custom section handling, and enhance Prettier file processing.

- Added mutex locking in ChangeCache and RollbackManager to prevent race conditions during manifest updates
- Updated gitignore logic to detect and preserve custom sections
- Enhanced Prettier batching and file formatting for better performance

2025-08-08 - 1.16.9 - fix(format)

Improve concurrency control in cache and rollback modules, refine gitignore custom section handling, and enhance Prettier file processing

- Added mutex locking in ChangeCache and RollbackManager to prevent race conditions during manifest updates
- Updated gitignore logic to detect and preserve existing custom sections from various markers
- Simplified Prettier formatter to process files sequentially, skip files without extensions, and log detailed status
- Minor refactoring in base formatter and tsconfig file updates for improved reliability

2025-08-08 - 1.16.8 - fix(format)

Improve concurrency control in cache and rollback management with mutex locking and refine formatting details

- Added 'withMutex' functions in ChangeCache and RollbackManager to synchronize file I/O operations
- Introduced static mutex maps to prevent race conditions during manifest updates
- Fixed minor formatting issues in commit info and package.json

2025-08-08 - 1.16.7 - fix(core)

Improve formatting, logging, and rollback integrity in core modules

- Add .claude/settings.local.json with defined permissions for allowed commands
- Standardize formatting in package.json, commit info, and configuration files
- Refactor rollback manager to use atomic manifest writes and validate manifest structure
- Enhance logging messages and overall code clarity in CLI and commit modules

2025-08-08 - 1.16.6 - fix(changecache)

Improve cache manifest validation and atomic file writes; add local settings and overrides

- Add manifest structure validation and default fallback in getManifest
- Implement atomic write in saveManifest using a temporary file and rename strategy
- Enhance error handling and cleanup for corrupted manifest files
- Introduce new `.claude/settings.local.json` for project-specific permission configuration
- Add an empty `assets/overrides.json` file for future overrides

2025-08-08 - 1.16.5 - fix(prettier)

Improve file selection in Prettier formatter, remove legacy package overrides, and update CI template indentation

- Added `.claude/settings.local.json` with updated permission settings for local commands
- Removed unnecessary overrides from `assets/overrides.json` and cleared `packageManager` overrides in `package.json`
- Adjusted CI template files (`ci_default_gitlab`, `ci_default_private_gitlab`, `ci_docker_gitlab`) for consistent indentation and formatting
- Refined Prettier formatter logic by defining include directories, root config files, and filtering duplicates instead of manual exclusion

2025-08-08 - 1.16.4 - fix(prettier)

Improve file exclusion in the Prettier formatter to skip unnecessary files and directories.

- Added exclusion patterns for `node_modules`, `.git`, `dist`, `.nogit`, `coverage`, `.nyc_output`, `vendor`, `bower_components`, `jspm_packages`, and minified files.
- Optimized filtering logic to ensure only valid files are processed.

2025-08-08 - 1.16.3 - fix(changecache/prettier)

Skip directories during file processing to prevent errors in changecache and prettier formatting

- Removed unnecessary await on synchronous file reads in changecache
- Added directory checks in changecache to immediately skip directories
- Filtered out directories in prettier formatter to avoid processing non-files

2025-08-07 - 1.16.2 - fix(format)

Fix format command confirmation prompt to correctly check user response

- Fixed bug where format command always showed "cancelled" even when user confirmed
- Changed response check from `response.proceed` to `response.value` for SmartInteract compatibility

2025-08-04 - 1.16.1 - fix(package/config)

Move smartdiff dependency to runtime and add local bash permissions settings

- Moved '@push.rocks/smartdiff' from devDependencies to dependencies in package.json
- Added `.claude/settings.local.json` with allowed bash commands (grep, mkdir, find, ls)

2025-05-19 - 1.16.0 - feat(format)

Enhance format module with rollback, diff reporting, and improved parallel execution

- Implemented rollback functionality with backup management and automatic rollback on error
- Added CLI commands for rollback, listing backups, and cleaning old backups
- Introduced DiffReporter for generating and displaying file diffs
- Improved file change caching via ChangeCache and expanded dependency analysis for parallel execution
- Updated logging to support verbose mode and enhanced user feedback
- Updated package.json to include new dependency '@push.rocks/smartdiff'

2025-05-14 - 1.15.5 - fix(dependencies)

Update @git.zone/tsdoc to ^1.5.0 and @types/node to ^22.15.18

- Bumped @git.zone/tsdoc from ^1.4.5 to ^1.5.0
- Bumped @types/node from ^22.15.17 to ^22.15.18

2025-05-13 - 1.15.4 - fix(package.json)

Update dependency versions: bump @git.zone/tsdoc, @push.rocks/lik, @push.rocks/smartlog, and @types/node to their latest releases

- Upgrade @git.zone/tsdoc from ^1.4.4 to ^1.4.5
- Upgrade @push.rocks/lik from ^6.0.15 to ^6.2.2
- Upgrade @push.rocks/smartlog from ^3.0.7 to ^3.0.9
- Upgrade @types/node from ^22.14.1 to ^22.15.17

2025-04-15 - 1.15.3 - fix(deps)

update dependency versions and improve website template variable handling

- Bumped @git.zone/tsbuild from ^2.2.1 to ^2.3.2 and @types/node to ^22.14.1
- Upgraded @push.rocks/smartscaf from ^4.0.15 to ^4.0.16 and prettier from ^3.5.2 to ^3.5.3
- Refactored website template update to correctly supply variables with added logging

2025-04-15 - 1.15.2 - fix(website_update)

Await supplyVariables call in website update template

- Changed website template update to properly await the supplyVariables method
- Ensured asynchronous consistency in updating website template variables

2025-04-15 - 1.15.1 - fix(cli)

Refresh internal CLI tooling and configuration for consistency.

2025-04-15 - 1.15.0 - feat(config/template)

Add assetbrokerUrl and legalUrl fields to module config and update website template to supply these values

- Added assetbrokerUrl and legalUrl properties in ts/classes.gitzoneconfig.ts
- Updated ts/mod_format/format.templates.ts to pass assetbrokerUrl and legalUrl to website template

2025-04-15 - 1.14.1 - fix(package.json)

Add packageManager field to specify pnpm version for consistent package management

- Inserted packageManager property in package.json with pnpm version info to ensure reproducible dependency installs

2025-04-15 - 1.14.0 - feat(tsconfig_update)

Add runafter directive to trigger gitzone format after tsconfig update

- Added runafter configuration in assets/templates/tsconfig_update/.smartscaf.yml to automate formatting task

2025-03-07 - 1.13.1 - fix(cli)

Improve commit message logging

- Updated logging to display recommended next commit details.
- Enabled interactive prompt for choosing commit type and scope.

2025-02-28 - 1.13.0 - feat(templates)

Updated and added new TypeScript template files for npm projects

- Added new paths.ts and plugins.ts template files for npm projects.
- Removed outdated some.plugins.ts template file.

2025-02-25 - 1.12.8 - fix(metadata)

Updated package and npmextra json description and keywords for enhanced development workflow clarity

- Updated the description in package.json to focus on project setup and management.
- Aligned the keywords in both package.json and npmextra.json to include more relevant terms such as gitzone utilities, template management, and CI/CD.

2025-02-25 - 1.12.7 - fix(meta)

Fix issues in project metadata and configuration.

- Updated package metadata to ensure accurate project description and licensing.

- Ensured npm access level configuration consistency within npmextra.json.

2025-02-25 - 1.12.7 - fix(ci)

Updated dependencies and added CI/CD workflows.

- Updated several dependencies in package.json for compatibility and security.
- Added default GitHub and GitLab CI/CD workflow files to the test directory.
- Introduced a .gitignore file to exclude certain directories and files.
- Ensured consistent formatting with Prettier and TypeScript configurations.

2025-01-29 - 1.12.6 - fix(project)

Minor fixes and cleanup

- Removed outdated pages/ directory entry in .gitignore.
- Adjusted Docker module files to include proper newlines.
- Updated TypeScript configuration for modular paths.
- Fixed formatting issues across various TypeScript files.

2025-01-29 - 1.12.5 - fix(cli)

Initial implementation of CLI utility with project management features

- Integration of various plugins for logging, command-line interactions, and project management.
- Setup CLI commands for project initialization, updating, formatting, and more.
- Implement commands for packaging, versioning, and deprecating npm packages.

2025-01-29 - 1.12.2 - fix(format)

Add overrides for peek-readable in package.json formatting

- Added a URL correction in the package.json repository information.
- Introduced support for pnpm overrides by including an `overrides.json` file.

2025-01-18 - 1.12.1 - fix(dependencies)

Update various package dependencies and Dockerfile base image

- Updated Dockerfile base image from 'alpinenpmci' to 'alpine_npmci'.
- Upgraded @git.zone/tsbuild, @git.zone/tsrun, @git.zone/tsdoc, and other dependencies to their latest versions.

2025-01-17 - 1.12.0 - feat(build)

Update TypeScript configuration to support emit decorator metadata

- Added emitDecoratorMetadata to the tsconfig.json template in assets/templates/tsconfig_update.

2025-01-08 - 1.11.0 - feat(cli)

Add Docker command for cleaning up Docker system and extend deprecation command for multiple registries

- Added a new command 'docker' to handle Docker system cleanup operations.
- Improved the 'deprecate' command to support deprecating packages across multiple npm registry URLs.

2025-01-01 - 1.10.10 - fix(templates)

Corrected typo in template file comment

- Fixed repeated comment in the template file for services under 'assets/templates/service/ts/some.plugins.ts'.

2025-01-01 - 1.10.9 - fix(templates)

Correct template file paths and organization for service projects

- Moved 'some.classes.some.ts' to 'classes.some.ts'
- Moved 'some.classes.some.db.ts' to 'classes.some.db.ts'
- Resolved incorrect import paths in service templates

2025-01-01 - 1.10.8 - fix(assets/templates)

Update CI template configurations to use module.github

- Replaced occurrences of `{{git.host}}` with `{{module.github}}` in CI workflow files
- Updated package dependencies for service template

2024-12-26 - 1.10.7 - fix(assets)

Correct URLs in templates and fix TypeScript declaration

- Updated incorrect URLs in Dockerfile templates to 'host.today'.
- Fixed type declaration for 'TemplateResult' in header.ts file.

2024-12-08 - 1.10.6 - fix(ci)

Corrected Docker image URL in CI templates

- Updated Docker image URL from 'code.foss.global/hosttoday' to 'code.foss.global/host.today' in default_nottags.yaml and default_tags.yaml.
- Adjusted gitignore template to include a custom section delineation.

2024-12-02 - 1.10.5 - fix(assets)

Update .gitignore template to remove pages directory

- Removed 'pages/' from the ignored directories in the .gitignore template.

2024-11-05 - 1.10.4 - fix(mod_format)

Correct file extension for TypeScript path configuration

- Fixed the TypeScript configuration to use correct file extensions for module subdirectories.

2024-10-27 - 1.10.3 - fix(mod_format)

Reorder TypeScript formatting steps in mod_format module

- Moved TypeScript configuration formatting earlier in the sequence for better logical consistency.

2024-10-27 - 1.10.2 - fix(format)

Add logging for tsconfig.json formatting

- Added an info log message for tsconfig.json formatting in format.tsconfig.ts.

2024-10-27 - 1.10.1 - fix(format)

Fixed async issue in tsconfig module lookup and corrected property access

2024-10-27 - 1.10.0 - feat(mod_format)

Add support for tsconfig.json formatting

- Added a new script to format tsconfig.json.
- Updated package.json to include `@git.zone/tspublish` as a dependency.

2024-10-23 - 1.9.126 - fix(format)

Remove redundant package.json property checks

- Removed property checks for `main`, `typings`, and `browserslist` from `format.packagejson.ts`
- This change streamlines the formatting process by removing unnecessary exits

2024-09-29 - 1.9.125 - fix(cli)

Fix package version configuration and formatting issues

- Updated metadata fields in package.json (repository URL, bugs URL, and homepage).
- Ensured presence and correctness of essential fields in package.json (main, typings, files, scripts, etc.).
- Added missing Prettier default TypeScript and Markdown configurations.

2024-09-27 - 1.9.124 - fix(cli)

Ensured proper existence and initialization of readme files

- Ensured `readme.md` and `readme.hints.md` files are created and initialized if they do not exist.

2024-09-27 - 1.9.123 - fix(core)

No changes detected

2024-09-27 - 1.9.123 - fix(core)

Update dependencies and improve build configurations

- Updated several dependencies in package.json for better compatibility
- Changed typedserver dependency version in website template
- Adjusted website server and service worker in templates
- Added security and test workflows for Gitea repositories
- Added and configured .gitignore, .npmrc, .vscode settings for the project
- Provided initial structure for readme and readme hints

2024-06-24 - 1.9.122 - fix(mod_commit)

Update package.json dependencies: @git.zone/tsdoc and @push.rocks/smartpromise to latest versions.

- ◦ Updated @git.zone/tsdoc to ^1.3.12
- ◦ Updated @push.rocks/smartfile to ^11.0.21

2024-06-23 - 1.9.121 - fix(mod_commit)

Fix changelog template rendering by removing extra new line when no version details are provided.

- Update package.json dependencies: @git.zone/tsdoc and @push.rocks/smartpromise to latest versions.

2024-06-23 - 1.9.120 - fix(mod_commit)

Handle edge case for empty version details in changelog formatting

- Added check for the length of the recommendedNextVersionDetails array
- Ensure no extra newline in changelog if there are no version details

2024-06-23 - 1.9.119 - fix(dependencies)

Update @git.zone/tsdoc to v1.3.8

- Updated @git.zone/tsdoc from v1.3.7 to v1.3.8 in package.json

2024-06-23 - 1.9.118 - fix(dependencies)

Update @git.zone/tsdoc to version 1.3.7

- Bump @git.zone/tsdoc from 1.3.6 to 1.3.7 in both package.json and pnpm-lock.yaml

2024-06-23 - 1.9.117 - fix(dependencies)

Update @git.zone/tsdoc dependency to v1.3.6

- Updated @git.zone/tsdoc version from 1.3.5 to 1.3.6 in package.json
- Updated pnpm-lock.yaml to reflect the new version of @git.zone/tsdoc

2024-06-23 - 1.9.116 - fix(dependencies)

Update @git.zone/tsdoc to version 1.3.5

- Updated the @git.zone/tsdoc dependency in package.json and pnpm-lock.yaml from version 1.3.4 to 1.3.5
- Removed the outdated changelog.md file.

2024-06-23 - 1.9.114 - fix(format)

Fixed formatting issues across multiple TypeScript files.

2024-06-23 - 1.9.113 - fix(mod_commit)

Remove extra new lines in changelog.

2024-06-23 - 1.9.112 - fix(core)

Update changelog formatting and remove outdated entries.

2024-06-23 - 1.9.111 - fix(changelog)

Remove outdated changelog entries and update formatting.

2024-06-23 - 1.9.110 -
fix(dependencies)

Update @git.zone/tsdoc to version 1.3.4.

2024-06-23 - 1.9.109 -
fix(changelog)

Remove outdated entries and adjust formatting in changelog.

2024-06-23 - 1.9.108 -
fix(dependencies)

Update @git.zone/tsdoc dependency to version 1.3.2.

2024-06-23 - 1.9.107 -
fix(changelog)

Remove placeholder entries and adjust formatting in changelog.

2024-06-23 - 1.9.106 -
fix(dependencies)

Updated @git.zone/tsdoc from version 1.3.0 to 1.3.1.

2024-06-23 - 1.9.105 - fix(dependencies)

Updated @git.zone/tsdoc dependency from 1.2.2 to 1.3.0 in package.json and pnpm-lock.yaml.

2024-06-23 - 1.9.104 - fix(changelog)

Remove placeholder entries and adjust formatting in changelog.

2024-06-23 - 1.9.103 - fix(changelog)

Fix changelog to remove placeholder entries and adjust formatting.

2024-06-23 - 1.9.102 - fix(logging)

Optimize logger instantiation and configuration.

2024-06-23 - 1.9.101 - fix(metadata)

Ensure accurate project metadata in package.json.

2024-06-23 - 1.9.100 - fix(dependencies)

Updated @git.zone/tsdoc dependency version to ^1.2.2 in package.json and pnpm-lock.yaml.

2024-06-23 - 1.9.99 - fix(mod_commit)

Fix variable reassignment issue in changelog writing step.

Revision #2

Created 2026-03-28 13:51:25 UTC by foss.global Team

Updated 2026-03-29 16:50:01 UTC by foss.global Team