

rpgit

native Git authority / bounded terminal workflows

User Guide

History, status, semantic diffs, safe mutations,
repository worktrees, and optional Codex workspaces

- Guide v0.2.0
- Debian 13 ARM64
- MIT OR Apache-2.0

GUIDE EDITION	SOURCE SNAPSHOT	DEBIAN PACKAGE
2026-08-24	91b2656	0.1.0-1apt.ryzhov.website
apt.ryzhov.website		github.com/ryzhov-artem/rgit

Contents

A practical guide to the current default interface, guarded Git workflows, configuration, remote operations, and the optional isolated Codex launcher.

1. What rpgit is	3
2. Install and verify	4
3. Launch and first workflow	5
4. Interface anatomy: History	6
5. Browse history and commit details	7
6. Review repository status	8
7. Understand the mutation safety model	9
8. Interface anatomy: Diff review	10
9. Work efficiently in Diff	11
10. Manage the repository family	12
11. Commands, configuration, and trust	13
12. External tools and remote operations	14
13. Advanced history and recovery	15
14. Optional isolated Codex workspace	16
15. Diagnostics and troubleshooting	17
16. Default keyboard quick reference	18
17. Default keyboard quick reference - review and management	19
18. Security, architecture, and engineering snapshot	20
References and project links	21

Default keys are configurable; press ? in any active view for the effective map and disabled reasons.

1. What rpgit is

rpgit is an original Rust terminal interface for Git, designed around responsive, bounded reads and explicit review before mutation. It presents repository history, status, diffs, branches, tags, stashes, worktrees, remotes, and selected history operations without replacing Git as the semantic authority.

Native Git remains authoritative

Repository reads, local mutations, authentication, transport, sequencer state, rebase, reflog, and bisect ultimately use native Git. The interface does not silently invent alternative repository semantics.

Review before mutation

Stage, unstage, commit, amend, discard, branch, stash, worktree, remote, and history actions first become immutable plans. Higher-risk actions require explicit or typed confirmation.

Bounded and responsive

History pages, process output, previews, caches, queues, scrollbar, and worker concurrency have explicit limits. Obsolete work is cancelled, and stale completions cannot replace the active selection.

Repository content is data

Repository-local configuration is inert until explicitly trusted. No background fetch, update check, telemetry, or provider call is permitted.

Current scope

Area	Current behavior
Repository discovery	Normal, bare, and linked-worktree repositories; explicit path or current directory.
Primary views	History, Status, Diff, Branches, Tags, Stashes, and Worktrees.
Local changes	Whole-file, whole-hunk, and selected-line workflows with preflight and confirmation.
Network	Only explicit typed remote commands; no automatic fetch.
Agent workflow	Optional isolated worktree and embedded Codex PTY through <code>rpgit-agent</code> .
Accessibility	Responsive layouts plus ASCII-only and no-color modes with textual markers.

Know the boundary

rpgit is a review and operation interface, not a replacement for understanding Git. Keep ordinary backups and read every destructive plan before confirming it.

Project references: [1], [2], [3]

2. Install and verify

The signed production package is published for Debian 13 (trixie), ARM64, including Raspberry Pi systems. The package installs both `/usr/bin/rpgit` and `/usr/bin/rpgit-agent`.

Install from the signed APT repository

```
sudo install -d -m 0755 /etc/apt/keyrings
curl -fsSL https://apt.rzhov.website/rdynamo.gpg \
| sudo tee /etc/apt/keyrings/rzhov-archive.gpg >/dev/null

sudo tee /etc/apt/sources.list.d/rzhov.sources >/dev/null <<'EOF'
Types: deb
URIs: https://apt.rzhov.website
Suites: trixie
Components: main
Architectures: arm64
Signed-By: /etc/apt/keyrings/rzhov-archive.gpg
EOF

sudo apt update
sudo apt install rpgit
```

Suite and architecture are intentional

Do not substitute another Debian suite or architecture unless the repository publishes a matching channel. The current production entry is `trixie / main / arm64`.

Verify the installation

```
rpgit --version
apt-cache policy rpgit
cd /path/to/a/git/repository
rpgit doctor
rpgit capabilities
rpgit-agent --help
```

Optional trust-anchor check:

```
gpg --show-keys --with-fingerprint \
/etc/apt/keyrings/rzhov-archive.gpg
```

Expected repository fingerprint: **6D8B 3463 113E 6804 658D 97CB 95CD 40A6 1168 5FEE**.

Build from source

```
git clone https://github.com/rzhov-artem/rgit.git
cd rgit
cargo build --release --locked

# Optional isolated Codex launcher:
cargo build --release --locked --features agent --bin rpgit-agent
```

The source baseline is Rust 2024 with MSRV 1.85. The default source build does not enable the PTY or agent feature; the Debian package contract includes the launcher.

Project references: [1], [2], [13]

3. Launch and first workflow

Open a repository

```
cd /path/to/repository
rpgit

# Or open an explicit repository / directory inside it:
rpgit /path/to/repository
```

rpgit draws the first terminal frame before repository work. Repository discovery and status/history reads then arrive as bounded events. The active repository may be a normal checkout, a bare repository, or a linked worktree.

Useful launch options

Option	Purpose
<code>--untracked no normal all</code>	Trade untracked-path detail for scan cost. Default: normal.
<code>--inline</code>	Render in an inline viewport instead of owning the alternate screen.
<code>--ascii</code>	Use ASCII-only graph and UI presentation.
<code>--no-color</code>	Disable color while retaining text, prefixes, rails, and line numbers.
<code>PATH</code>	Open this repository or a directory inside it.

The first 60 seconds

- **j** / **k** moves the selected history commit.
- **s** opens Status; **h** returns to History.
- **v** opens a semantic diff for the selected commit or tracked path.
- **Space** marks paths in Status; **a** stages and **u** unstages after plan review.
- **c** opens the internal commit editor; **Ctrl-S** validates and opens the commit plan.
- **?** opens generated help; **:** opens the typed command line; **q** exits.

Recommended terminal

An 80 x 24 terminal is a comfortable starting point. The interface has tested compact/drawer/wide layouts, including much shorter or wider geometries, and degrades to ASCII/no-color without losing semantic markers.

Project references: [1], [3], [5], [12]

4. Interface anatomy: History

The default History surface combines repository identity, a bounded commit graph, selected commit details, progress state, and context-sensitive key hints.

```

rpgit firecracker / main 10 0 changes
HISTORY [FOCUS] [STALE] · 100
1 ● 6e7cb66ae115 devices: Remove blanket #![allow(unused)] 21h
  ● b40f757b3fbf pci: Drop the unused device lock in allocate_bars() 21h
  ● 061b3e842ea4 devices: Handle errors that were being silently dropped 2d
  ● 87c1bc7bc6aa pci: Fail snapshot restore if device activation fails 2d
  ● 8f6e93257b82 balloon: Compare queue indices when dispatching test eve... 2d
  ● 9c493b4dad4 test: x86: update T2CL MSR baselines 2d
  ● 4aa9bb83c1c1 tests: x86: verify CPUID-gated MSR passthrough 2d
  ● d9edec6f4646 vmm: x86: remove obsolete CPU configuration path 2d
  ● 4681a67f712a vmm: x86: fix CPUID-dependent MSR read ordering 2d
  ● b851857505c7 vmm: x86: vcpu: split boot configuration stages 2d
  ● d46a5a210d30 vmm: x86: cpu_config: split CPU template application sta... 2d
  ● 6c5d9f3e0417 feat(snapshot): allow overriding huge-page configuration 11d
  ● b1577f905f47 test: restore vulnerability finding subset comparison 2d
  ● af94a06a5059 test: replace vulnerability exception regexes with predi... 4d
  ● b1a2fcb44c66 test: require guest kernel version for vulnerability exc... 4d
  ● 8a45f30150a8 test: clarify SRS0 guest exceptions 4d
  ● aba6c645f449 test: update T2S MSR fingerprints for FB_CLEAR 4d
  ● 255b7d69101e test: remove T2S MMIO Stale Data exceptions 4d
  ● 05d47d0f8ab1 cpu-template: set FB_CLEAR in T2S 4d
  ● 33ba9eb7e7dc test: remove BPI guest exceptions 4d
  ● 6a3715ebfd19 test: remove REPTAR guest exception 4d
  ● 5312192fef3f test: pass host microcode version to guests without CPU ... 4d
  ● 03cfdeba34bb test: run spectre-meltdown-checker with VMM mode on host 4d
  ● 5f6e856c5e8c tests: Add integration tests for virtio-pci BAR relocat... 20d
  ● 116cfe0f4ca3 pci: Add unit tests for BAR relocation 5d
  ● ce4e7b14f5da pci: Support virtio-pci BAR relocation 20d
  ● 542cc37a9963 seccomp: Allow KVM_IOEVENTFD on the vCPU thread for BAR... 20d
  ● 01d42743778e pci: Free the device BAR range on detach 2d
  ● d9d04a5a56a3 pci: Free the device BAR range on detach 2d
  ● 20de4a8ea920 Merge branch 'main' into dependabot/cargo/firecracker-d... now
HISTORY | j/k select / filter n next page Tab next view ? act 5
  
```

#	Surface	Meaning
1	Repository header	Repository, attached branch or detached object, upstream distance, change/conflict counts, and active state.
2	History graph	Bounded commit rows with topology, object prefix, subject, decorations, and relative age.
3	Commit details	Subject, author, full object identity, parent count, decorations, committer, timestamp, encoding, signature, and message.
4	Progress rail	Current bounded refresh/loading state without blocking navigation.
5	Mode footer	Priority-fitted keys for the active view; ? remains the authoritative help route.

Project references: [4], [12]

5. Browse history and commit details

History is loaded in pages of 100 commits by default. Status, history, decorations, and commit details are independent bounded tasks, so a slow detail read does not freeze selection or status rendering. No history navigation contacts a remote.

Key	Action
j / Down	Select next commit; prefetch near a page boundary.
k / Up	Select previous commit.
Home / End	Jump to first or final retained commit.
n	Load or retry the next bounded history page.
/	Edit a literal, fixed-string history filter.
d	Retry details for the selected commit.
y	Send the selected full object ID through OSC 52.
o	Restore the terminal and open controlled <code>git show</code> .
v	Open selected commit against its first parent in Diff.
r	Refresh repository-local data under a new epoch.

Filtering and graph stability

The history filter becomes one independent `--grep=...` argument with `--fixed-strings`; it is never shell text. Continuations are tied to the exact repository epoch, query, retained count, and boundary object. If refs move across a page boundary, rpgit reloads page one instead of silently duplicating or omitting rows.

Bytes and terminal safety

Object IDs, parent lists, names, subjects, and messages remain byte-oriented at the Git boundary. Invalid bytes and terminal control characters are rendered as inert escapes, not emitted as terminal instructions.

Clipboard behavior

OSC 52 is a request to the surrounding terminal. SSH, tmux, and terminal policy decide whether the copied object ID reaches your local clipboard; rpgit cannot force acceptance.

Project references: [4]

6. Review repository status

Status uses byte-safe Git porcelain-v2 data and groups visible paths by conflict, staged, unstaged, and untracked state. Selection follows the visible grouping while retaining exact native path bytes for operations.

Status navigation and selection

Key	Action
j / k, arrows	Move through visible paths.
Home / End	Move to first or final visible path.
Space	Mark or unmark the current exact path.
v	Open the selected tracked path in Diff.
a	Build a stage plan for marked eligible paths, or the current path.
u	Build an unstage plan.
c	Open the internal multiline editor for a regular commit.
A	Open explicit amend.
e / E	Choose Git-configured, Vim, or nano for commit/amend.
D	Build a guarded discard plan.
l	Show the latest bounded, redacted operation journal record.
p / P	Stash tracked changes / include untracked changes.

Internal commit editor

Input	Editor behavior
Enter	Insert newline.
Tab	Insert four spaces.
Backspace	Edit draft.
Ctrl-S	Validate the message and open plan review.
Esc	Cancel without running Git.

Commit drafts are passed through standard input and are omitted from debug state and journal records. External editors receive a complete terminal handoff; rpgit reacquires and redraws the terminal afterward.

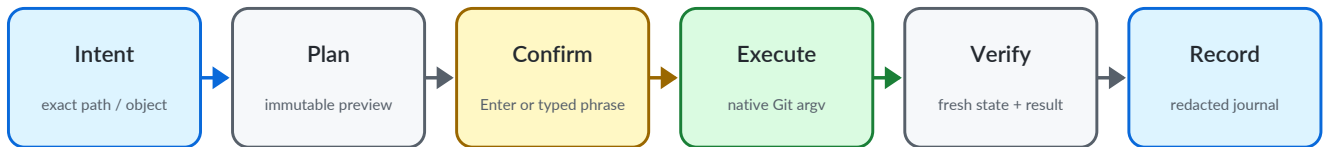
Large worktrees

Use `--untracked no` to avoid untracked discovery, or `--untracked all` for full detail. The default `normal` mode keeps directory-level discovery bounded.

Project references: [5], [6]

7. Understand the mutation safety model

Every supported mutation follows the same visible contract: typed intent becomes an immutable plan, fresh preconditions are checked, the user confirms, native Git executes, and repository state is verified and refreshed.



Repository data is re-read before safety-sensitive execution; no displayed path is parsed back into Git.

Confirmation classes

Operation	Confirmation
Stage / unstage / regular commit	Review the plan, then press Enter .
Amend / one-path discard	Review the explicit warning, then press Enter .
Multi-path discard	Type the exact displayed phrase discard N .
Force branch deletion, stash drop, worktree prune, force push, hard reset, published rewrite	Typed confirmation binds the exact target shown in the plan.

The operation journal

The private journal lives under `.git/rpgit/journal/`. It records bounded metadata such as operation type, argv and stdin byte counts, precondition totals, result status, refresh outcome, and recovery guidance. It deliberately excludes paths, patches, commit messages, process output, environment values, and credentials.

Hunk and selected-line preflight

Before a hunk or selected-line patch can reach confirmation, rpgit hashes and checks it with native Git. Immediately before execution it re-reads the diff, regenerates the selection and fingerprint, and requires exact byte agreement. Stale, binary, mode-only, unsupported, or incomplete cases fail closed.

Esc versus execution

Before the execution boundary, **Esc** cancels the plan. After native Git has started a defined operation, rpgit waits for its verified result instead of pretending that cancellation restored an unknown state.

Project references: [3], [6], [7]

8. Interface anatomy: Diff review

The semantic diff view keeps file selection, target identity, stable line gutters, additions/deletions, bounded position, and operation hints visible while content scrolls.

```

rpgit firecracker / main 10 10 0 changes
FILES · 37
M src/vmm/src/devices/acpi/vmclock.rs [COMMIT] [SYNTAX Rust]
modified 100644 -> 100644 [AUTO->UNIFIED: narrow]
@@ -1,7 +1,6 @@
1 1 // Copyright 2025 Amazon.com, Inc. or its affiliates. All Rights Reserved.
2 2 // SPDX-License-Identifier: Apache-2.0
3 3
4 4 - use std::convert::Infallible;
5 4 use std::mem::offset_of;
6 6 use std::sync::atomic::{Ordering, fence};
7 6
@@ -94,7 +93,7 @@ impl VmClock {
94 93 EventFd::new(libc::EFD_NONBLOCK).map_err(VmClockError::CreateEventFd)?,
95 94 );
96 95
97 - let mut inner = vmclock_abi {
98 96 + let inner = vmclock_abi {
99 97     magic: VMLOCK_MAGIC,
100 98     size: VMLOCK_SIZE,
101 99     version: 1,
102 99 @@ -180,7 +179,7 @@ impl<'a> Persist<'a> for VmClock {
180 179 }
181 180 }
182 181
183 - fn restore(vm: Self::ConstructorArgs, state: &Self::State) ->
Result<Self, Self::Error> {
182 182 + fn restore(_: Self::ConstructorArgs, state: &Self::State) ->
Result<Self, Self::Error> {
184 183     let interrupt_evt = EventFdTrigger::new(
185 184 EventFd::new(libc::EFD_NONBLOCK).map_err(VmClockError::CreateEventFd)?,
DIFF | a/u stage/unstage hunk i worktree/index j/k scroll [/] hunk / search ? actions

```

#	Surface	Meaning
1	File list	Changed files and exact path identity; the selected file remains visually stable.
2	Target header	Selected path, commit/range/worktree/index target, syntax mode, and truncation state.
3	Stable gutters	Selection rail, old/new line numbers, separator, and change kind stay fixed during horizontal scrolling.
4	Semantic rows	Added/deleted/context rows with prefixes, background, optional syntax and word emphasis.
5	Scrollbar	Bounded vertical position in long unified or side-by-side content.
6	Action footer	Hunk staging, worktree/index switching, navigation, search, and generated help.

Project references: [7], [12]

9. Work efficiently in Diff

Press **v** from Status to inspect a tracked path, or from History to inspect a commit against its first parent. The title labels the target as [WORKTREE], [INDEX], [COMMIT], or [RANGE].

Key	Action
j / k, arrows	Scroll vertically.
Left / Right	Scroll long content horizontally without losing gutters.
{ / }	Previous / next file.
[/]	Previous / next hunk.
i	Switch mutable preview between worktree and index.
S	Cycle automatic, side-by-side, and unified presentation.
/, n, N	Literal search; next / previous match.
+	Request the advertised larger bounded preview after soft truncation.
a / u	Review whole-hunk stage / unstage.
B / H	Open bounded blame / rename-following path history.
P / R	Commit-parent comparison / recent resolved comparisons.
E	Preview optional external viewer handoff.

Selected-line staging

Key	Action
J / K	Move the changed-line cursor.
Space	Toggle one changed line.
V	Anchor or complete a range.
Ctrl-x	Clear selected lines.
A	Submit the non-empty immutable preview for native Git preflight.

Presentation is optional; semantics are not

Automatic side-by-side appears only when the measured content panel is wide enough and the file has a normal bounded textual diff. Narrow terminals, binary data, combined forms, oversized alignment work, and other unsupported cases fall back visibly to unified rendering. Syntax and word emphasis are bounded presentation layers; line prefixes, line numbers, hunk rails, and target identity remain authoritative.

External viewer

The internal semantic renderer remains the fallback. **E** previews a configured `review-viewer` or a successfully probed `delta`. Exact patch bytes, argv, working directory, environment policy, and input size are shown before terminal handoff.

Project references: [7]

10. Manage the repository family

rpgit treats a repository as one shared object store plus every main, linked, detached, missing, locked, or prunable checkout reported by Git. Direct view keys keep each repository surface one keystroke away.

Key	View	Primary actions
b	Branches	C create, S switch, R rename, X merged delete, F typed force-delete, W linked worktree.
t	Tags	L lightweight, T annotated, G signed, D local delete.
x	Stashes	a apply, o pop, D typed drop.
w	Worktrees	R remove clean linked checkout, P dry-run and typed prune.

Branch ownership

A local branch owned by any checkout cannot be switched to, renamed from the wrong checkout, or deleted as if it were free. The Branch view exposes ownership together with upstream distance.

Worktree path safety

- Creation resolves and validates the target parent, rejects symlink targets, and accepts only a missing path or an empty real directory.
- The target may not contain or nest inside another checkout and may not be inside the common Git directory.
- Removal refuses the main/current checkout, locked or missing entries, and any tracked, untracked, ignored, or conflicted contents.
- rpgit calls `git worktree remove` without force and never recursively deletes checkout contents itself.

Stash conflicts are ordinary Git conflicts

Apply and pop may stop with conflicts. rpgit refreshes and reports the native result; it does not retry, auto-resolve, or claim that pop removed the stash unless the refreshed reflog proves it.

Project references: [8]

11. Commands, configuration, and trust

One typed command registry drives keys, the : command line, completion, generated help, availability reasons, and safety classes. This prevents key handling and command behavior from drifting apart.

?

Generated help for the current view, including the effective key sequence, source, safety class, and disabled reason.

:

Typed command line with context-aware Tab completion and bounded, memory-only history.

/

Literal search in the active view. Enter accepts, Esc cancels, n/N navigate.

!

Configured external commands. Nothing runs before an immutable preview is confirmed.

Configuration locations and precedence

```
$XDG_CONFIG_HOME/rpgit/config.toml
$XDG_STATE_HOME/rpgit/trust-v1
<worktree>/.rpgit.toml
```

built-in defaults -> user -> trusted repository -> CLI -> session

Without XDG variables, user configuration defaults to `~/.config/rpgit/config.toml` and trust state to `~/.local/state/rpgit/trust-v1`.

Minimal version 1 configuration

```
version = 1
untracked = "all"
key_sequence_timeout_ms = 500
history_page_size = 100

[[keymap]]
keys = "g g"
command = "selection.last"
context = "global"
mode = "normal"
```

Repository trust

A repository `.rpgit.toml` file is parsed with strict bounds but remains inert until the repository is explicitly trusted. The header reports `CONFIG UNTRUSTED`, `CONFIG TRUSTED`, or `CONFIG INVALID`.

```
:trust.grant
:trust.revoke

rpgit config path
rpgit config check
rpgit config explain
```

Trust is outside the checkout

Trust is keyed by the canonical common Git directory plus object format. Committing a configuration file into the repository cannot grant itself execution authority.

12. External tools and remote operations

Prefer executable plus argv configuration

```
version = 1

[[command]]
name = "inspect-path"
argv = ["printf", "%s\\n", "{selected_path}"]
working_directory = "{repo}"
environment = ["TOOL_MODE=inspect"]
```

Each argv value remains one process argument. Supported whole-value placeholders include {repo}, {worktree}, {git_dir}, {common_git_dir}, {selected_path}, {commit}, and {branch}. Embedded or unknown brace expressions fail closed.

Shell mode is deliberately conspicuous

A command with `shell = true` shows HIGH RISK on every run and requires the exact typed phrase **run shell NAME**. There is no persistent bypass.

Remote commands are always explicit

```
:remote.fetch [remote]
:remote.pull-ff
:remote.pull-merge
:remote.pull-rebase
:remote.push [remote]
:remote.push-upstream [remote]
:remote.force-with-lease [remote]
```

Startup, refresh, focus, timers, and repository watching never fetch. Pull uses a separately fetched local observation and a selected strategy rather than an implicit configuration-dependent `git pull`. Push shows the exact source and destination. Force push is only force-with-lease against an expected object ID and requires typed confirmation.

Authentication handoff

```
:remote.fetch-terminal [remote]
:remote.push-terminal [remote]
:remote.force-with-lease-terminal [remote]
```

Terminal variants completely restore the terminal and let native Git use the user's credential helpers, SSH agent, pinentry, and signing facilities. rpgit does not store credentials.

13. Advanced history and recovery

Create a recovery anchor first

Rebase and reflog reset require an application-owned backup of the current HEAD. Start with **:history.backup**, review its exact refs/rpgit/backups/. . . name, and retain it until recovery is no longer needed.

Exact-object operations

```
:history.cherry-pick <full-commit-oid>
:history.revert <full-commit-oid>
:history.continue
:history.skip
:history.abort
```

Object arguments are full SHA-1 or SHA-256 IDs. Abbreviations, arbitrary revision expressions, and editable shell scripts are rejected by the planner. Native Git owns sequencer state; rpgit observes and exposes only supported, separately reviewed transitions.

Typed rebase

```
:history.backup
:rebase.pick <full-upstream-oid>
:rebase.plan <full-upstream-oid> \
  pick:<oid>,reword:<oid> "new subject"
```

Supported plan actions are pick, reword, edit, squash, fixup, and drop. The validated sequence must contain every commit in the chosen range exactly once. A published rewrite requires the displayed typed phrase and never implies a push.

Reflog and bisect

```
:reflog.show
:reflog.branch <full-oid> <new-branch>
:reflog.reset-mixed <full-oid>
:reflog.reset-hard <full-oid>

:bisect.start <full-good-oid> <full-bad-oid>
:bisect.good
:bisect.bad
:bisect.skip
:bisect.reset
```

Recovery checklist

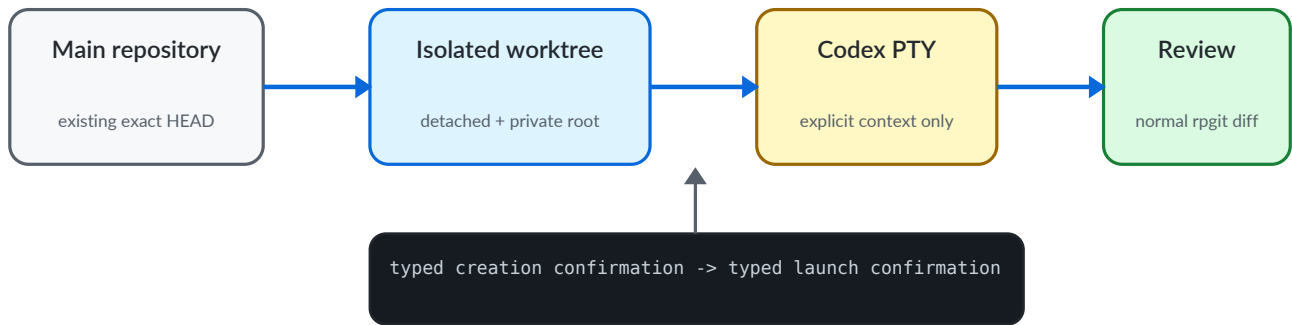
- Read the active operation label and conflict count in the header.
- Inspect Status without starting an incompatible second mutation.
- Resolve and stage conflicts, then review continue; otherwise review skip or abort when available.
- After an unwanted rewrite, create a branch at the displayed backup ref instead of guessing a reflog selector.
- Inspect the latest redacted result through `ℓ` or the journal command.

Project references: [10]

14. Optional isolated Codex workspace

rpgit-agent creates an isolated detached worktree, previews the exact workspace and context, requires typed confirmation before creation and before provider launch, embeds Codex in a bounded PTY, and returns resulting files to the ordinary rpgit review flow.

Credentials remain in the provider's normal environment/configuration.



Start the helper

```

mkdir -m 0700 -p /absolute/private/rpgit-workspaces
cd /path/to/repository
rpgit-agent --workspace-root /absolute/private/rpgit-workspaces

# Optional explicit context files; repeat --context:
rpgit-agent \
  --workspace-root /absolute/private/rpgit-workspaces \
  --context README.md \
  --context src/main.rs
  
```

The workspace root must already exist. Context paths must be normal repository-relative files; they are opt-in, bounded, and may not escape the repository. Codex itself must be installed and authenticated separately. The package never contains Codex credentials.

PTY ownership modes

Mode / key	Behavior
[PTY]	Keyboard and paste go to Codex. Ctrl-\ always returns to the host.
[HOST] Enter	Return focus to a running child.
[HOST] q	Request bounded child shutdown.
PageUp / PageDown	Navigate retained scrollbar.
/, n, N	Search parsed retained cells.
v + arrows/hjkl	Select parsed cells; left-mouse drag is also supported.
y	Send sanitized selected text, or the visible grid, through bounded OSC 52.
after exit: Enter	Open the isolated checkout in ordinary rpgit review.
after exit: x / r	Close while retaining workspace / rebuild launch preview and reconfirm.

Review remains the boundary

Agent output never bypasses the normal Status, Diff, immutable operation plans, or Git mutation confirmations. Dirty or unrecognized workspaces are not automatically deleted.

Project references: [1], [11]

15. Diagnostics and troubleshooting

Non-interactive diagnostics

```
rpgit config path [--json]
rpgit config check [--json]
rpgit config explain [--json]
rpgit doctor [--json]
rpgit capabilities [--json]
rpgit benchmark --iterations 10000 [--json]
```

These commands do not require a TTY. JSON uses a versioned management schema and is suitable for scripts or package verification.

Common symptoms

Symptom	Check / action
"not a repository" or discovery failure	Run <code>git rev-parse --show-toplevel</code> and pass an explicit repository path to rpgit.
Capabilities reports unsupported Git	Install Git 2.35 or newer; stash headers in porcelain-v2 require it.
Repository config has no effect	Inspect <code>rpgit config check</code> , then review and run <code>:trust.grant</code> .
OSC 52 copy does not reach the desktop	Check terminal, SSH, and tmux clipboard policy. The UI reports only that the request was sent.
Vim or nano plan fails before confirmation	Install the editor or use Git-configured mode; rpgit performs a bounded version preflight.
Remote operation needs credentials	Use the reviewed <code>*-terminal</code> variant so native Git owns the interactive terminal.
Large worktree scans slowly	Start with <code>--untracked no</code> or the default <code>normal</code> instead of <code>all</code> .
Terminal presentation is difficult to read	Use <code>--ascii</code> , <code>--no-color</code> , or <code>--inline</code> ; press <code>?</code> for current hints.
Interrupted sequence after restart	Refresh, read the active Git operation in the header, then review continue/skip/abort.
Agent child appears to own input	Press <code>Ctrl-\</code> ; both legacy and enhanced keyboard forms are recognized as the host escape.

When reporting an issue

- Include `rpgit --version`, `git --version`, operating system, architecture, terminal emulator, and whether SSH/tmux is involved.
- Attach `rpgit doctor --json` after reviewing it for environment details.
- Describe the active view, exact action, visible plan/result, and whether the repository is normal, bare, or a linked worktree.
- Do not publish credentials, private paths, commit contents, or repository-local secrets.

Project references: [5], [6], [9], [10], [11]

16. Default keyboard quick reference

Generated help wins

Configuration can remap or unbind keys. Press **?** in the active view for the effective key sequence, source, safety class, and disabled reason.

Global and view navigation

Key	Action
j / k, arrows	Move selection or scroll in the active surface.
Home / End	First / final retained item or row.
Tab	Next repository view.
h / s / v / b / t / x / w	History / Status / Diff / Branches / Tags / Stashes / Worktrees.
r	Refresh repository state.
/	Literal search in the active view.
n / N	Next / previous search match where available.
:	Typed command line with Tab completion.
!	Configured external command list and preview.
?	Generated help.
Ctrl-Z	Suspend on Unix with terminal restoration.
q	Exit or close the active read-only surface.

History

Key	Action
n	Load / retry next page.
d	Retry commit details.
y	Copy full object ID through OSC 52.
o	Controlled native <code>git show</code> handoff.
v	Open selected commit diff.

Status and commit

Key	Action
Space	Mark/unmark exact path.
a / u	Plan whole-file stage / unstage.
c / A	Internal regular commit / explicit amend.
e / E	External editor chooser for commit / amend.
D	Guarded discard plan.
l	Latest redacted journal result.
p / P	Stash tracked / include untracked.

17. Default keyboard quick reference - review and management

Diff

Key	Action
{ / }	Previous / next file.
[/]	Previous / next hunk.
i	Worktree / index target.
S	Automatic / side-by-side / unified.
Left / Right	Horizontal content scroll.
+	Larger bounded preview after soft truncation.
a / u	Whole-hunk stage / unstage.
J / K	Move changed-line cursor.
Space / V / Ctrl-x / A	Toggle line / range / clear / submit selected-line preview.
B / H	Blame / path history.
P / R	Parent comparison / recent comparisons.
E	Optional external viewer preview.

Branches, tags, stashes, worktrees

View	Keys
Branches	C create, S switch, R rename, X merged delete, F force-delete, W linked worktree.
Tags	L lightweight, T annotated, G signed, D delete.
Stashes	a apply, o pop, D typed drop.
Worktrees	R safe remove, P dry-run and typed prune.

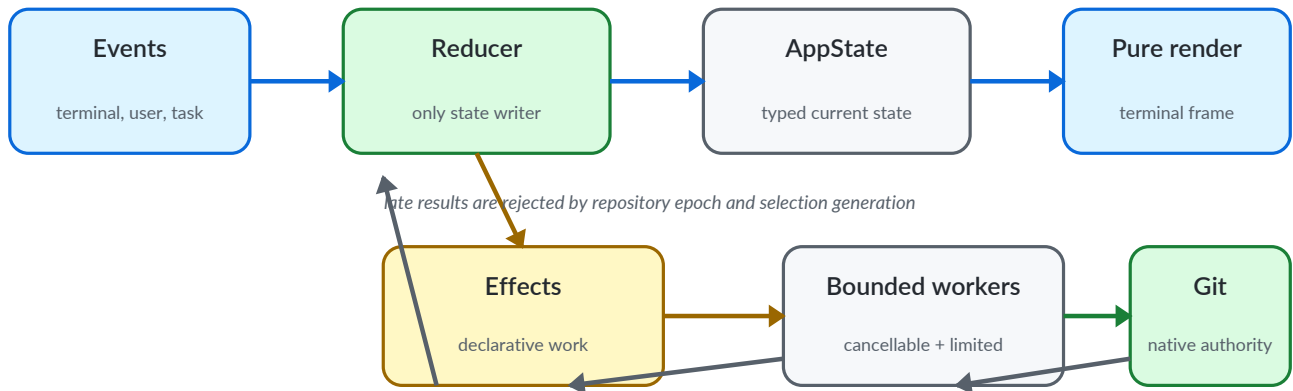
Agent PTY inspection

Key	Action
Ctrl-\	Return from [PTY] to [HOST].
Enter	Return to running child; after exit, open review.
q	Bounded child shutdown while running.
j/k, Ctrl-u/d, PgUp/PgDn, g/G	Scroll one row, half-page, page, oldest/live.
/, n, N	Search retained parsed cells.
v, arrows/hjkl, mouse drag	Create and extend parsed-cell selection.
y	Sanitized OSC 52 copy.
after exit: x / r	Close and retain workspace / rebuild preview and reconfirm.

Project references: [1], [4], [6], [7], [8], [11]

18. Security, architecture, and engineering snapshot

The project separates application state, effects, Git boundaries, and pure terminal presentation. The reducer is the only application-state writer; bounded workers report typed completion events, and stale results are rejected before they can alter the current view.



Security and privacy boundaries

- No shell is inserted around ordinary Git commands; executable and argv remain separate.
- Filesystem paths and machine-readable Git data stay native bytes until display; rendered strings never flow back into filesystem or Git operations.
- Credentials, environment secrets, clipboard bytes, pasted text, and private-key material are excluded from logs and the operation journal.
- No background fetch, telemetry, update check, or provider call.
- External commands require a visible preview; shell mode requires a high-risk typed phrase every time.
- Optional in-process `gix-read` remains experimental/diagnostic; production repository semantics and all mutations stay on native Git.

Engineering snapshot

Property	Snapshot
Language / edition	Rust 2024; minimum supported Rust 1.85.
Terminal stack	Ratatui 0.29.0 and Crossterm 0.28.1.
Unsafe Rust	<code>unsafe_code = "deny"</code> .
Primary package target	Debian 13 / AArch64; Raspberry Pi 5 is a native validation target.
Source baseline	Linux and macOS.
License	MIT OR Apache-2.0, at the user's option.
UI evidence snapshot	Repository-surface matrix across seven views, multiple terminal geometries, dark/light/high-contrast/mono/ASCII/no-color policies, plus native Raspberry Pi/tmux validation.

Design principle

Review first. Mutate deliberately. Keep native Git authoritative.

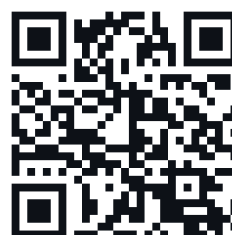
Project references: [2], [3], [12]

References and project links

Ref	Source
[1]	Project README https://github.com/ryzhov-artem/rgit/blob/main/README.md
[2]	Cargo package metadata https://github.com/ryzhov-artem/rgit/blob/main/Cargo.toml
[3]	Architecture contract https://github.com/ryzhov-artem/rgit/blob/main/docs/architecture.md
[4]	History browser https://github.com/ryzhov-artem/rgit/blob/main/README.md#navigation
[5]	Status browser https://github.com/ryzhov-artem/rgit/blob/main/README.md#highlights
[6]	Mutation workflows https://github.com/ryzhov-artem/rgit/blob/main/README.md#design
[7]	Diff viewer https://github.com/ryzhov-artem/rgit/blob/main/README.md#highlights
[8]	Repository management https://github.com/ryzhov-artem/rgit/blob/main/README.md#highlights
[9]	Command system, config, trust, and remotes https://github.com/ryzhov-artem/rgit/blob/main/README.md#build-and-run
[10]	Advanced history operations https://github.com/ryzhov-artem/rgit/blob/main/README.md#highlights
[11]	Embedded PTY and agent workspace https://github.com/ryzhov-artem/rgit/blob/main/docs/agent.md
[12]	UI improvement Step 3 evidence https://github.com/ryzhov-artem/rgit/blob/main/docs/development.md#fixtures-and-terminal-checks
[13]	Signed APT repository https://apt.ryzhov.website/



Signed packages
apt.ryzhov.website



Source and issues
github.com/ryzhov-artem/rgit

Document scope

Document revision **0.2.0**. This guide describes the installed rpgit 0.1.0 commands, Debian package **0.1.0-1apt.ryzhov.website**, and source snapshot **91b2656**. Public source and issue tracking are published exclusively at **github.com/ryzhov-artem/rgit**; the installed commands remain **rpgit** and **rpgit-agent**. Later releases may add, remove, or remap behavior.

Screenshots supplied by the project maintainer. Diff screenshot terminal chrome was cropped to remove machine/session identifiers before publication.