Unfaithfulness in the Axiom Stacks formalization

HumanLean Team

November 2026

What counts as unfaithful?

A formal proof establishes the claim written in code. Faithfulness asks whether that claim and its proof preserve the mathematics of the source text. We assess two dimensions separately:

- **Definition and statement faithfulness:** the formalization preserves the intended objects, assumptions and conclusions. A separating example can expose a mismatch by satisfying one definition but failing the other.
- **Proof faithfulness:** the formalization preserves the substantive argument of the source proof. Routine changes in notation or implementation do not count. Replacing the central argument with another method counts as a proof difference, even when both methods prove the same theorem correctly.

The audit concerns Axiom-written formalizations. Direct registrations of existing Mathlib results and transparent wrappers around them are outside its scope. Findings must concern the tagged definition, statement or its own proof. Differences inherited only from dependencies are excluded. Harmless generalizations and explicitly disclosed partial coverage are also excluded.

Findings

Across five foundational chapters—Set Theory, Categories, Topology, Sheaves on Spaces, and Sites and Sheaves—we identify **94 definitions or results with substantive differences**:

- **6** definition mismatches.
- **1** statement mismatch.
- **88** substantive proof replacements.

One result has both a statement and a proof difference. The seven definition or statement findings concern what the formalization expresses. The proof findings concern how it establishes its conclusion.

The findings below are organized by chapter. Detailed analyses explain the six definition mismatches through separating examples and the statement mismatch through its missing compatibility conditions. Every finding links the source argument and formalization supporting the comparison.

Chapter 3. Set Theory

000J — Proof replacement

In the documented cardinal model, the formal proof chooses a beth strong-limit cutoff of uncountable cofinality rather than iterating least-rank representative and diagram closure. Original · Formalization · Source lines · Code lines · Code lines 2.

04T6 — Proof replacement

The size inequality agrees. The formal proof replaces the finite base-cover reduction by a compatible-basic-open cover, small-basis counting and finite-restriction estimates. Original · Formalization · Source lines · Code lines · Code lines 2.

000R — Proof replacement

In the declared cardinal model, the finite-type closure proof establishes a direct size inequality by inverse-image affine covers instead of the source's (5)→(7)→(6) route. Other clauses include faithful specializations and generalized coproduct bounds. Original · Formalization · Source lines · Code lines.

0AHK — Proof replacement

The theorem agrees. The formal descent step uses injectivity of faithfully flat stalk maps, cancellation through the factors, and germ separation. The source uses global-section injection and factorization. Original · Formalization · Source lines · Code lines · Code lines 2.

000X — Proof replacement

Within the explicitly cardinal-only model, the formal proof uses arbitrary support representatives and direct cardinal sums/site closure rather than least-rank representatives, transfinite rank closure and high cofinality. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

Chapter 4. Categories

04AT — Proof replacement

The formal proof keeps the original connected finite diagram, first equalizes its endomorphisms, then grows a partial limit along adjacent objects. The source instead reduces to a bipartite diagram and inducts by merging source vertices with pullbacks. Both establish the same finite-connected-limit criterion. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

09WR — Proof replacement

The generation argument uses a retraction onto a generated subgroup and common-upper-bound closure induction. The source uses a direct-sum quotient. The assertion agrees after the formalization's explicit nonempty correction. Original · Formalization · Source lines · Code lines.

0A1S — Proof replacement

Direct witness and coherence construction replace the source appeal to the Hom-colimit characterization of essential constancy. Original · Formalization · Source lines · Code lines.

0A1T — Proof replacement

A connectedness-induction construction replaces the displayed Hom-colimit/opposite-invariance deduction. Original · Formalization · Source lines · Code lines.

0FWY — Proof replacement

A canonical-comma/Kan extension construction replaces the chosen-presentation construction. The compact factorization and independence arguments remain source-aligned. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

05Q2 — Proof replacement

The formal proof reverses the displayed deduction direction, using an elementwise refinement argument instead of invoking Hom-colimit/finite-limit interchange. The mathematical principle agrees. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

05Q6 — Proof replacement

For finite-limit preservation under right localization, common denominators and simultaneous refinements construct compatible cone lifts, followed by coyoneda reflection. The displayed source proof instead dualizes finite-colimit preservation under left localization. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

04YR — Proof replacement

On the explicitly chosen canonical model, the formal proof constructs a direct projection inverse. The source instead reassociates fibre products and erases repeated factors. The statement agrees on that model. Original · Formalization · Source lines · Code lines · Code lines 2.

034H — Proof replacement

For part (2), the formal proof constructs an inertia lift by lifting an automorphism and its inverse along a cartesian arrow. The source deduces fibredness from the diagonal 2-pullback and equivalence transport. The diagonal equivalence in part (1) follows the source. Original · Formalization · Source lines · Code lines.

04Z7 — Proof replacement

For the diagonal criterion, paired total-category arrows, lifts of base maps and vertical comparison isomorphisms establish fullness and faithfulness directly. The source instead applies the fibrewise equivalence criterion and characterizes the comparison on each fibre. The assertion agrees. Original · Formalization · Source lines · Code lines · Code lines 2.

02XT — Proof replacement

The formal proof transfers the two groupoid-fibration axioms directly through the slice forgetful functor. The displayed source argument instead uses the coproduct of fibres. Original · Formalization · Source lines · Code lines.

09WW — Proof replacement

Successive applications of the two primitive groupoid-fibration axioms replace the displayed deduction through fibredness and invertibility in the fibres. The source itself suggests this alternative route. This finding compares the substantive argument with the displayed proof. Original · Formalization · Source lines · Code lines.

06N7 — Proof replacement

For fibredness over the total base, the formal proof lifts only in X, chooses the new Y-object as the image of that lift, and uses an identity comparison. The source invokes the general two-fibre-product argument with two component lifts. The witness and projection-to-Y conclusion agree. Original · Formalization · Source lines · Code lines · Code lines 2.

0045 — Proof replacement

The quotient-category construction uses a presheaf of isomorphism classes and admissible base arrows, replacing the source descent-of-composition construction. Its fullness and faithfulness arguments identify the same mathematical category. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

02Y3 — Proof replacement

For uniqueness of the representing object, transport of terminal identity objects replaces the source presheaf/Yoneda argument. This concerns the object-isomorphism assertion covered by the formal statement. Original · Formalization · Source lines · Code lines.

0FFM — Proof replacement

The central reverse-invertibility argument reflects a single object isomorphism by full faithfulness, while the source identifies a whole quasi-inverse functor. The three-condition theorem agrees. Original · Formalization · Source lines · Code lines.

003Q — Definition mismatch: uniqueness of compatible 2-cells

Intuition. Treating routes as the same up to an identification can conceal two different identifications of a route with itself. The original definition requires uniqueness of that identification. Original · Formalization.

Original claim. For every square Q, the groupoid of cone morphisms $Q \to P$ is nonempty, and between any two such morphisms there is exactly one compatible invertible 2-cell. A 2-cell compares morphisms. A groupoid is a category in which every morphism is invertible. Source finality and 2-fibre-product definition.

Formal claim. Each set of isomorphism classes of cone morphisms is a singleton. The code first quotients cone morphisms by the existence of a compatible 2-isomorphism, then asks for ordinary terminality. Formal relation, quotient homs and terminality predicate.

Separating example and why it works

Use the strict 2-category on the diamond $P < x, y < z$, with one 1-cell for each comparable pair. Every nonempty hom-groupoid is terminal except $\mathrm{Hom}(P, P) = B(C_2)$: its sole 1-cell is id_P, with two invertible 2-endomorphisms. Here C_2 is the two-element group. Horizontal composition at P is addition in C_2. Whiskering into another hom-category sends both elements to its identity 2-cell.

For the cospan $x \to z \leftarrow y$, only P admits arrows to both x and y. Thus there is one square, one cone pre-morphism id_P with its leg comparisons, and one quotient hom class. The formal square category is terminal. Both elements of C_2 are compatible automorphisms of that same pre-morphism because both leg whiskerings kill them. The source uniqueness condition fails.

The example satisfies the ambient assumptions: its 2-category is strict and every 2-cell is invertible. The discrepancy comes from discarding the distinction between the two compatible automorphisms.

06N8 — Statement mismatch and proof replacement: base-relative compatibility

Statement. The formal theorem omits compatibility conditions required by the original result. Original · Formalization.

The original result constructs an equivalence strictly over the base Y with the required vertical comparison over C, and in the commuting case over Y. The tagged theorem proves an ordinary categorical equivalence with natural isomorphisms. Its conclusion omits strict over-base and vertical-coherence requirements. Formal comparison theorem.

Proof. Its cancellation argument replaces the source's common-base-relative-pullback construction. Source argument and formal cancellation proof.

The code comments say the generalized result specializes fully to 06N8. Additional lifting or strictification may recover the stronger conclusion under the source hypotheses. The missing step is to supply those conditions and connect them to the stated equivalence. The explicit over-base equivalence structure records the stronger compatibility data.

Chapter 5. Topology

0GM3 — Proof replacement

Growing a connected component union until one component remains replaces selecting an endpoint of a maximum-distance graph pair. Original · Formalization · Source lines · Code lines.

04MF — Proof replacement

An entire connected component, made clopen using finitely many closed components, replaces the source's union of irreducible components through the point. Original · Formalization · Source lines · Code lines.

08ZM — Proof replacement

Choosing closed specializations and transferring cluster points replaces the uncovered-closed-remainder contradiction and ambient finite subcover. Original · Formalization · Source lines · Code lines.

09UV — Proof replacement

A finite original subcover is padded by empty sets to become point-finite, then shrunk by Mathlib's normal-space shrinking lemma, instead of selecting and grouping pointwise small neighbourhoods. Original · Formalization · Source lines · Code lines.

09UW — Proof replacement

Simultaneous finite tuple choices replace the nested arity/bad-tuple induction and cover-member splitting. Original · Formalization · Source lines · Code lines · Code lines 2.

09UX — Proof replacement

Simultaneous compact tuple separations replace induction on arity and successive deletion of closed bad sets from selected members. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

0CY5 — Proof replacement

A full-arity shrink operator modifying participating coordinates replaces lower-arity repetition handling and single-member bad-set removal. Original · Formalization · Source lines · Code lines · Code lines 2.

0A2P — Proof replacement

Canonical interior–kernel-image cylinders, obtained from directed neighbourhood filters of the initial limit topology, replace explicit finite product boxes; the compact case applies directed-cover elimination to those cylinders. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

08ZV — Proof replacement

TopCat.compactSpace_of_isLimit uses monadic lifting to CompHaus instead of the source's closed-equalizer subspace of a compact product. Original · Formalization · Source lines · Code lines.

0540 — Proof replacement

Density of the complement and a contradictory dense-open intersection replace constructing and gluing a neighbourhood inside the subset. Original · Formalization · Source lines · Code lines · Code lines 2.

0541 — Proof replacement

An irredundant irreducible cover of closure of the complement replaces the source's constructibility and point-neighborhood criteria. Original · Formalization · Source lines · Code lines · Code lines 2.

06NA — Proof replacement

Direct specialization lifting from openness and finite π-fibres replaces the source relation-space/saturation descent; the strengthened theorem is disclosed and valid. Original · Formalization · Source lines · Code lines.

02IB — Proof replacement

Finite-gap induction establishes specialization invariance of the difference of two dimension functions. The source uses the codimension formula. The common-generic-point neighbourhood agrees. Original · Formalization · Source lines · Code lines.

09XW — Proof replacement

The coordinate embedding is proved directly from the compact-open basis, rather than via the source spectral-bijection and specialization-lifting criterion; the statement agrees. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

0A31 — Proof replacement

For conclusion (a), spectrality is built directly from ambient generic points and patch-compact intersections rather than a spectral inverse-limit theorem. The other conclusions are faithful. Original · Formalization · Source lines · Code lines · Code lines 2.

0AP0 — Proof replacement

After identifying the limit, the formal argument obtains spectrality from patch-closedness of the whole intersection. The source uses a stagewise-spectral inverse-limit argument. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

0909 — Proof replacement

Encode points by closures of images of domain opens instead of traces of regular closed domains on the dense image; the cardinality bound agrees. Original · Formalization · Source lines · Code lines.

0B20 — Proof replacement

For part (b), preservation of limits by the forgetful functor to groups is derived from a composite functor and reflection of limits. The source instead reduces to products and equalizers and checks their coordinatewise group operations and induced topology. Original · Formalization · Source lines · Code lines · Code lines 2.

0B21 — Proof replacement

Colimit existence: construct the algebraic colimit with the finest compatible group topology instead of solution-set bounds and representability. Forgetful preservation follows the source adjunction. Original · Formalization · Source lines · Code lines.

0B24 — Proof replacement

Existence: build the algebraic commutative-ring colimit with a coinduced lattice ring topology instead of solution-set bounds and representability. Preservation uses the source indiscrete-topology mechanism. Original · Formalization · Source lines · Code lines.

09Y0 and 09Y1 — Definition mismatches: locally closed strata

Intuition. A stratum is a layer of a space. Requiring each layer to be locally closed means it is open within its closure. The code permits layers interwoven densely throughout the same space. Original 09Y0 · Formalization 09Y0 · Original 09Y1 · Formalization 09Y1.

Original claims. The section-local definition of a partition requires locally closed parts. A good stratification, 09Y0, adds the frontier condition: a stratum meeting another's closure lies in that closure. A stratification, 09Y1, adds a partial order with each stratum's closure contained in the union of lower strata. Source partition convention, good stratifications and stratifications.

Formal claims. The two structures require disjoint covering subsets and the respective closure conditions. Their complete fields omit local closedness. Good-stratification fields and stratification fields.

Two separating examples

09Y0. In $X = \mathbb{R}$ take the two strata $\mathbb{Q}$ and $\mathbb{R} \setminus \mathbb{Q}$. They are nonempty, disjoint, cover X, and both have closure X. Every frontier implication therefore holds. A dense locally closed subset is open in its closure, hence open in X. Neither subset is open, so neither is locally closed. The formal fields hold and the original definition fails.

09Y1. In $X = \mathbb{R}$ take a top index T with stratum $S_T = \mathbb{Q}$ and one index i_r for each irrational r, with stratum $S_{i_r} = \{r\}$. Order every i_r below T, retaining reflexivity and no other comparisons. This is a partial order. The lower union for T is all of X, which equals the closure of $\mathbb{Q}$. For i_r, its lower union and closure are both $\{r\}$. All formal closure bounds hold. The rational stratum is dense and not open, so is not locally closed.

Why the distinction matters. The primary definition immediately preceding the tagged source definitions explicitly requires locally closed parts. Neither structure includes an equivalent requirement in its other fields.

Chapter 6. Sheaves on Spaces

0098 — Proof replacement

Two linked implementations use substantive skyscraper-adjunction/representability proofs instead of the source tensor-stalk calculation. The linked tensor-model implementation follows the source route; all assert the expected stalk formula. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

009N — Proof replacement

Lean constructs the extension as compatible sections on all basis subopens and proves it a sheaf by per-basis gluing, instead of the source locally representable stalk-family construction. The extension theorem agrees. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

009Q — Proof replacement

Lean proves the elementary basis-sheaf/site bridge and uses dense-subsite/right-Kan equivalence descent, rather than the source cover-indexed colimit of stalk fibre products. The extension statement agrees. Original · Formalization · Source lines · Code lines · Code lines 2.

009T — Proof replacement

For existence and canonicity, the formal proof descends a module through the basis restriction equivalence. The source instead constructs pointwise multiplication on germ families. Original · Formalization · Source lines · Code lines · Code lines 2.

009W — Proof replacement

Lean directly glues images over inverse-image basis covers and verifies linearity/naturality, instead of transporting a restricted morphism through the source basis restriction equivalence. The f-map extension theorem agrees. Original · Formalization · Source lines · Code lines · Code lines 2.

009X — Proof replacement

Lean directly glues C-valued arrows over basis covers rather than constructing set-valued germ families and lifting algebraic structure through stalk colimits. The f-map theorem agrees, at slightly greater generality. Original · Formalization · Source lines · Code lines · Code lines 2.

00A0 — Proof replacement

In clauses (4)–(5), full faithfulness is proved using adjunctions on opens instead of direct evaluation of sections. The formalization explicitly covers sheaves of sets. Original · Formalization · Source lines · Code lines · Code lines 2.

00A3 — Proof replacement

For clause (5), Lean builds an open-immersion adjoint triple and transfers counit invertibility to unit invertibility, replacing the source section-evaluation proof. The extension/restriction and stalk statements agree. Original · Formalization · Source lines · Code lines · Code lines 2.

00AE — Proof replacement

The counit proof derives its isomorphism from categorical full faithfulness. The source uses the stalkwise-isomorphism criterion. The covered assertions agree. Original · Formalization · Source lines · Code lines · Code lines 2.

Chapter 7. Sites and Sheaves

00VW — Proof replacement

Generated-sieve equality replaces isomorphism cancellation and the explicit pullback-section argument; the statement is a faithful generalization. Original · Formalization · Source lines · Code lines.

00VX — Proof replacement

Generated-topology inclusion and sheaf monotonicity replace the source per-family refinement and pullback-section comparisons. Original · Formalization · Source lines · Code lines.

00VY — Proof replacement

The proof still selects cross-representatives, but concludes through equality of presieve collections and generated topologies, rather than sheaf-comparison lemmas. Original · Formalization · Source lines · Code lines.

00W0 — Proof replacement

Gluing on all regular-object arrows replaces orbit reduction and stabilizer fixed-point calculations. The group statement is covered by a disclosed monoid generalization. Original · Formalization · Source lines · Code lines · Code lines 2.

0H6Y — Proof replacement

The formal proof verifies local injectivity and surjectivity on covering objects, then invokes the sheafification recognition criterion. The source constructs compatible sheafification representatives. Original · Formalization · Source lines · Code lines.

086K — Proof replacement

Direct local-lift gluing replaces extension from the presheaf image through its sheafification. Original · Formalization · Source lines · Code lines · Code lines 2.

00WT — Proof replacement

Explicit local coproduct preimages replace the source sheafified-Yoneda Hom-injectivity test. Original · Formalization · Source lines · Code lines.

08H2 — Proof replacement

An auxiliary subterminal support presheaf and adjunction replace extracting an arrow from a left-Kan-colimit representative. Original · Formalization · Source lines · Code lines · Code lines 2.

0D05 — Proof replacement

To separate quasi-compactness from finite refinement, a chain-defined covering system and maximal-rung argument replace the source ordinary-open-cover compactness example. The existential conclusions agree. Original · Formalization · Source lines · Code lines · Code lines 2.

0GMR — Proof replacement

In part (4), terminal quasi-compactness selects a surjective single-stage fibre before K is chosen in S; the source chooses K over the whole fibre coproduct before finite extraction. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

0A35 — Proof replacement

Adjunction and a colimit universal property identify an existing inverse image before applying its sections theorem, instead of constructing the source pointwise-colimit sheaf first. Exactness is deduced afterwards. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

09YN — Proof replacement

Closed-relation classifiers and subsheaf generation/gluing replace the source double-colimit and fixed-site sections argument. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

0EXJ — Proof replacement

Direct descent of arbitrary compatible morphisms replaces deriving the colimit bijection from 09YN and a constant sections diagram. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3 · Code lines 4.

00XM — Proof replacement

Under the explicit 00XL transfer premise, invertibility of the sheafified unit is transferred instead of using the source Hom/Yoneda comparison. This file does not derive transfer from cocontinuity. Original · Formalization · Source lines · Code lines.

03CD — Proof replacement

An explicit sigma-presheaf/restriction adjunction and uniqueness of left adjoints replace the source's component-initial Kan-colimit computation; the sheaf formula agrees. Original · Formalization · Source lines · Code lines · Code lines 2.

00Y1 — Proof replacement

In the linked 00Y1 equivalence, the sigma/fibre presheaf constructions agree with the source, but local-bijectivity and fibre-faithfulness arguments replace transport through the source's sheafification-commuting square. Original · Formalization · Source lines · Code lines · Code lines 2.

04IL — Proof replacement

For 04IL(2), the linked implementation reconstructs a factorization-presheaf adjunction to prove post-composition, instead of invoking lower-shriek composition. Its proof of (1) by uniqueness of right adjoints is explicitly offered by the source. Original · Formalization · Source lines · Code lines · Code lines 2.

04TR — Proof replacement

For essential surjectivity of descent-data gluing, a cover-dense total-category construction and chosen preimage replace the source's direct compatible-tuple section construction; the gluing result agrees. Original · Formalization · Source lines · Code lines · Code lines 2.

0EYV — Proof replacement

The inverse-image base-change isomorphism is proved through a pullback-factorization/Guitart-exact right-Kan argument. The source uses lower-shriek functors and a sheafified-representable fibre product. Original · Formalization · Source lines · Code lines · Code lines 2.

03CF — Proof replacement

In the general 03CF implementation, adjoint uniqueness and a separate connected-comma terminal/pullback exactness proof replace the source's sheafification-cancellation order. Clause (d) agrees. Clause (c) uses yonedaIso, with its identification with 04D3 explicitly left unproved. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

03CI — Proof replacement

A site of subsheaves of finite-product trees and image factorizations replaces iterative fibre-product/subsheaf closure followed by a union; the stated presentation properties agree. Original · Formalization · Source lines · Code lines · Code lines 2.

03A2 — Proof replacement

A generator-only intermediate sieve site and sheafness transport through f* replace the source's iteratively fibre-product-closed site and covering-first continuity argument; its final Yoneda/adjunction comparison agrees. Original · Formalization · Source lines · Code lines · Code lines 2.

04H1 — Proof replacement

The two localization-square comparisons use finite-product naturality and uniqueness of right adjoints directly. The source reduces to good sites and representables, then transports the result. Original · Formalization · Source lines · Code lines.

00YA — Proof replacement

For clause (2), cofiltered-elements and dense-Yoneda reconstruction replace verifying the original point axioms followed by transport of stalk exactness. The canonical germ comparison and density extension follow the source. Original · Formalization · Source lines · Code lines · Code lines 2.

04EL — Proof replacement

For 04EL's continuity step, singleton section tests and cofiltered common refinements replace the source's covering-fibre-product verification. Its singleton-cover equivalence and Map(u(U),E) push-forward computation agree. Original · Formalization · Source lines · Code lines · Code lines 2.

0F4F — Proof replacement

The formal comparison identifies composite left adjoints to the same explicit right adjoint, instead of flattening the iterated colimit and checking connected fibers. The stalk formula agrees. Original · Formalization · Source lines · Code lines · Code lines 2.

04H2 — Proof replacement

An elements-category initiality argument compares stalk colimits directly, instead of the source auxiliary set-site and composite-topos construction. Original · Formalization · Source lines · Code lines.

04H6 — Proof replacement

The formal proof detects covering sieves through stalk-surjectivity, instead of using the localization equivalence and coproduct decomposition of stalks. Original · Formalization · Source lines · Code lines.

0F4G — Proof replacement

The formal stalk comparison is obtained from equal skyscraper right adjoints, rather than the source presheaf-pushforward and sheafification-invariance factorization. Original · Formalization · Source lines · Code lines.

00YP — Proof replacement

A set-sized application of Zorn's lemma to partial covering choices and finite obstructions replaces the source well-ordered recursion and unions of actual refinement systems. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

00YQ — Proof replacement

The formal construction separates closed-sieve classifier sections through an explicit omega tower and proves conservativity by sieve detection, instead of separating arbitrary sheaf sections by the source argument. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

04D7 — Proof replacement

The formal theorem assumes operational C-cover section lifts; the source derives them from the sheafified-representable coproduct factorization. The equivalence bridge is explained but not formally constructed here. Original · Formalization · Source lines · Code lines · Code lines 2.

04D9 — Proof replacement

An argument later in the same file derives reflection of epimorphisms from faithfulness and categorical cancellation. It replaces the source two-level local-lifting argument. The tag marks a supporting step. Original · Formalization · Source lines · Code lines.

04B6 — Proof replacement

The formal empty-image-sieve argument uses local surjectivity of the general sheafification unit, rather than the source separated-empty one-plus calculation. Original · Formalization · Source lines · Code lines · Code lines 2.

08LW — Proof replacement

The formal proof transfers full faithfulness across the localization adjoint triple and uses explicit slice morphisms, instead of the source good-site reduction and representable/Yoneda calculation. Original · Formalization · Source lines · Code lines · Code lines 2 · Code lines 3.

00ZO — Proof replacement

The formal proof reduces to local bijectivity and an explicit image-sieve/pullback calculation, instead of constructing the identity-sieve class and a Yoneda inverse. Original · Formalization · Source lines · Code lines.

00ZQ — Proof replacement

The formal proof reduces to the join topology and equality of sheaf categories. The source uses a successive-sheafification diagram. Original · Formalization · Source lines · Code lines.

00WV — Definition mismatch: image families and covering sieves

Intuition. A site specifies which families of maps count as coverings. Adding every smaller map that factors through a member can change a permitted family into a family that is not selected as a covering. Original · Formalization.

Original claim. A continuous functor sends each selected covering family to a selected covering family with the same indices, and preserves the required covering base-change pullbacks. Source covering and pullback clauses.

Formal claim. `IsContinuous.cover_pushforward` asks that the entire factorization closure of the image family be a selected pretopology covering. A pretopology specifies covering families. Its generated Grothendieck topology specifies covering sieves, which are families closed under precomposition. This distinction matters for the coded field. Tagged field and factorization closure.

Separating example and why it works

Take the thin category $0 < 1$, with its trivial pretopology: only singleton isomorphism families cover. Thin means there is at most one arrow between any two objects. Use the identity functor.

The identity preserves every selected family and all pullbacks, so it satisfies the original definition. Pullbacks in this category are minima. At object 1, the family $\{\mathrm{id}_1\}$ is a selected covering. Its factorization closure contains both $0 \to 1$ and id_1. That two-arrow family is not a singleton isomorphism family, so it fails the coded covering condition. This gives a source-continuous functor that fails the tagged predicate.

Why the distinction matters. Saturation is natural when working with Grothendieck topologies, but this field uses unsaturated `Pretopology` coverings. Unfolding `functorPushforward` confirms factorization closure. The separate untagged `MapsCoveringsToCoverings` clause uses the faithful raw image family. Its existence does not change the tagged full definition. Trivial-pretopology coverings and faithful raw-image clause.

Explicit source warning. The formalization explicitly acknowledges this mismatch and retains the tag because changing the definition would affect approximately ninety files. A faithful covering-preservation clause exists separately, but the tagged definition still uses the changed clause. Explicit source warning.

08LX — Definition mismatch: the ambient image of an open subtopos

Intuition. An abstract category can sit inside a larger category in different ways. Knowing that both placements preserve all maps does not show that their images contain the same objects. Original · Formalization.

Original claim. A strictly full subcategory $E \subset \mathrm{Sh}(C)$ is an open subtopos when it is the essential image of the localization pushforward $j_{F,*}$ from $\mathrm{Sh}(C)/F$, for a subterminal sheaf F. Subterminal means that F is a subobject of the terminal object. The essential image includes objects isomorphic to the functor's values. Source embedding convention, subtopos image and open subtopos.

Formal claim. `IsOpenSubtopos J P` requires an equivalence of the full subcategory selected by P with `Over F`, compatible with its domain-forgetful functor. This identifies the ambient inclusion with $j_{F!}$. Formal ambient-inclusion convention and tagged definition.

Separating example and why it works

Work in the category of sets, realized as sheaves on the one-object trivial site. Take $F = \varnothing$ and let $P(X)$ mean $X \cong \varnothing$. The slice $\mathrm{Set}/\varnothing$ has only the empty-domain object, up to unique isomorphism, so it is a terminal category. Its equivalence with the full subcategory of empty sets commutes with domain-forgetful. Thus the coded predicate holds.

A geometric-embedding pushforward is a right adjoint and preserves the terminal object. Its essential image must therefore contain the singleton set 1. The subcategory of empty sets excludes 1, so cannot be a source subtopos. Concretely, for this empty slice, $j_{F!}$ has image the empty sets and $j_{F,*}$ has image the singleton sets.

Why the distinction matters. The neighboring 08LW correctly shows both outer adjoints are fully faithful for subterminal F. That property concerns their maps and does not equate their essential images. Source definition 08LV and the library's `ToposLocalization` both use the pushforward for the embedding image, so swapped notation does not resolve the difference. The formal comments expressly choose the domain-forgetful inclusion. Library localization adjoints and full-faithfulness comparison.

090L — Definition mismatch: enough objects with an arbitrary property

Intuition. A family of objects with a property P can generate a covering sieve even when it is not a selected covering family. Refining to a selected family can lose P. Original · Formalization.

Original claim. A site has enough P objects when every object U admits a selected covering family $\{U_i \to U\}$ with $P(U_i)$ for every i. Source clause (3).

Formal claim. `HasEnoughObjects J P` requires such a family whose generated sieve belongs to the Grothendieck topology J. It asks for coverage by the generated sieve, independently of membership in the original selected basis. Formal generated-sieve clause.

Separating example and why it works

Take the full category of finite sets on the two objects $A = \{0\}$ and $B = \{0, 1\}$. Select only singleton isomorphism coverings. These form a site: isomorphisms compose and base change. Let P mean cardinality two.

There is no selected P-cover of A. Its selected covering arrows must be isomorphisms onto A, and hence have singleton domains.

The generated Grothendieck topology contains exactly the maximal covering sieve at each object. The map $p : B \to A$ has a section $s : A \to B$. Since $p \circ s = \mathrm{id}_A$, the sieve generated by p contains the identity and is maximal. It therefore covers A using a P-domain. The identity of B supplies a P-cover of B. These witness the formal enough-P predicate for every object, while the original enough-P predicate fails at A.

Why the distinction matters. The definitions agree if coverings are taken to be the maximal pretopology associated with J. The source's raw sites do not require this saturation. Mathlib distinguishes a pretopology from the largest pretopology generating its topology. Source site axioms and generated topology versus maximal pretopology.

Scope. This finding concerns clause (3), which allows an arbitrary property P.

Sources and verification

The comparisons inspect the formal declarations, surrounding definitions and comments, and the proof dependencies needed to interpret them. Separating examples are mathematical arguments; selected finite examples were additionally checked in Python. They have not been compiled as Lean counterexamples.

The audit uses Axiom Stacks revision `5ed82d1`, its pinned Mathlib revision `56a0058`, and Stacks Project revision `a04446e`. The cited prose agrees in mathematical content with the source revision declared by the formalization. Links throughout the report identify the exact passages supporting each comparison.

The Original and Formalization links open the tag pages. The revision-pinned Source lines and Code lines identify the versions assessed here.