Article

Contribution level licensing prepares the FathomNet Database for new annotation types



GitHub: github.com/dbold23

e-mail: daniel.sam-bold@gmail.com

Code: fathomnet-py

Data: database.fathomnet.org

Daniel Sambold[1]

The FathomNet Database is a public repository of annotated underwater imagery, built on FAIR data principles to open marine imagery to everyone and to build a catalog of marine life. Until its v2.6.0 release in September 2026, its licensing terms were recorded only as a sentence on each of 346 upload batches, such as "Images licensed as CC BY-NC (URL), annotations licensed as CC BY (URL)", with no queryable terms on any image or annotation. As the database grows toward annotation types beyond bounding boxes, and toward compatibility with other databases and with Creative Commons rules, those terms need to be readable by software. We propose recording licensing at the level of the contribution: every image carries an image license and an annotation license, in separate columns that each hold one of seven Creative Commons identifiers in our proposed schema. Both are needed, because the image license and the annotation license differ on 240,952 of 472,639 images (51%). We implemented the model as seven database migrations and verified it against a replica of production. None of the migrations was merged upstream as written; the one merged pull request is a separate access control repair. The maintainers instead carried our license columns and backfill into their own release (v2.6.0), which now stores an image license on each image and an annotation license on each bounding box, and are moving annotations toward a simpler, single table design, currently a draft proposal. With new annotation types storable in the proposed schema, a proof of concept run tested how they could be filled: SAM 2 derived 1,169,279 masks from the database's boxes, and an automatic grader, not yet checked against a reference set, accepted 74.4% of them. Under a proposed derivation rule, which is not implemented and is not a legal finding, masks would be held back on the 240,912 images (51%) whose image license forbids sharing derivatives; no mask has been ingested. A comparison with SAM 3, which differs from SAM 2 in its image encoder among other components, finds the two in close agreement on the median (IoU about 0.8) and suggests a split by animal shape at the margins.

Image databases have become infrastructure for ocean science. The FathomNet Database is a public repository of underwater imagery and expert annotations built on FAIR data principles[1], so that marine imagery is findable, accessible, interoperable and reusable by anyone[2]. Its aim is a catalog of all marine life; machine learning is one use of that catalog but not the audience it was originally built for. In the replica of production copied on 28 July 2026, it held 472,639 images and roughly 1.28 million bounding box annotations. Researchers contribute imagery and annotations in batches, and anyone can browse and download them.

Reuse depends on licensing, and that is where this work picks up. In the July 2026 snapshot used here, the terms on every image were stored in one free text column on the Darwin Core[3] record of each upload batch, as a sentence such as "Images licensed as CC BY-NC (URL), annotations licensed as CC BY (URL)". No image row and no annotation row carried terms, and nothing could be queried on them. A researcher asking whether 200 images may be used to train a commercial model had to open each image, follow it to its parent upload and read the sentence. A contributor who drew boxes on someone else's photograph had no place to state terms for their own work. Making the terms machine readable is the next step toward compatibility with other databases and with Creative Commons rules.

Licensing also decides how the database can grow to new annotation types, such as points and segmentation masks. The contributed corpus is a monolith of bounding boxes. Masks can be derived from those boxes at scale with promptable segmentation models[4], but a derived mask is a new contribution built on another party's image and another party's box, so whether it may exist, and under what terms, is a licensing question before it is a modeling one.

Here we describe a licensing model at the level of the contribution, stated as four design principles that can be checked against any annotation corpus. We implement it as seven database migrations, verify it against a replica of production, and show that image level and annotation level terms differ on 51% of images, which shows that images and annotations need separate license values (P1). The migrations were not merged upstream, and the Discussion records what the maintainers did instead. We then report a proof of concept run that derives 1.17 million SAM 2 masks from the existing boxes, compare SAM 2 with SAM 3, whose image encoder, among other components, differs, and record the simpler direction the maintainers chose.

Licensing was one sentence per upload batch

A contributor uploads a batch, which owns one Darwin Core row and links to many images through a many to many join table (Fig. 1); 5,378 images belong to two or more collections, so their terms come from more than one sentence. Until the maintainers' v2.6.0 release (September 2026), the only rights columns were three on `darwin_cores`: `license`, `rights_holder` and `access_rights`; images and bounding boxes had no license column, the backend

[1]Monterey Bay Aquarium Research Institute (MBARI), Moss Landing, California, USA.

source had no license type and the API had no license endpoint. The license column held a sentence that the upload form built by joining two dropdown choices. 245 collections created before the upload interface (through August 2025) shared one identical string covering 212,795 images, about 45% of everything. Each batch carried its own copy of a sentence, but few sentences were distinct: the 346 rows hold at most 102, and the live database in October 2026 holds five (Methods).

Three defects follow, and they need different fixes. The granularity is wrong, because an image reached through several collections can inherit terms that disagree. Each sentence makes two claims, terms for the images and terms for the annotations, and the two genuinely differ (see below). And the prose is still being written: until v2.6.0 the CSV upload contract carried license terms only as text inside the Darwin Core string, and v2.6.0 added structured image and annotation license fields but still accepts and stores the prose. Downstream, the official Python client's COCO export stamps one hard-coded license, pointing at a URL that returns a 404 page, onto every dataset it produces (still so in v1.10.2).

A fourth problem is structural. Before v2.6.0 the bounding box row had nineteen columns, of which four are geometry; the rest identify the row, state a claim (the concept), name the claimant (the observer) and record review. The schema does not separate a scientific claim from its shape, so the row is the unit of counting, reviewing and exporting, and since v2.6.0, which added a license column to it, of licensing as well.

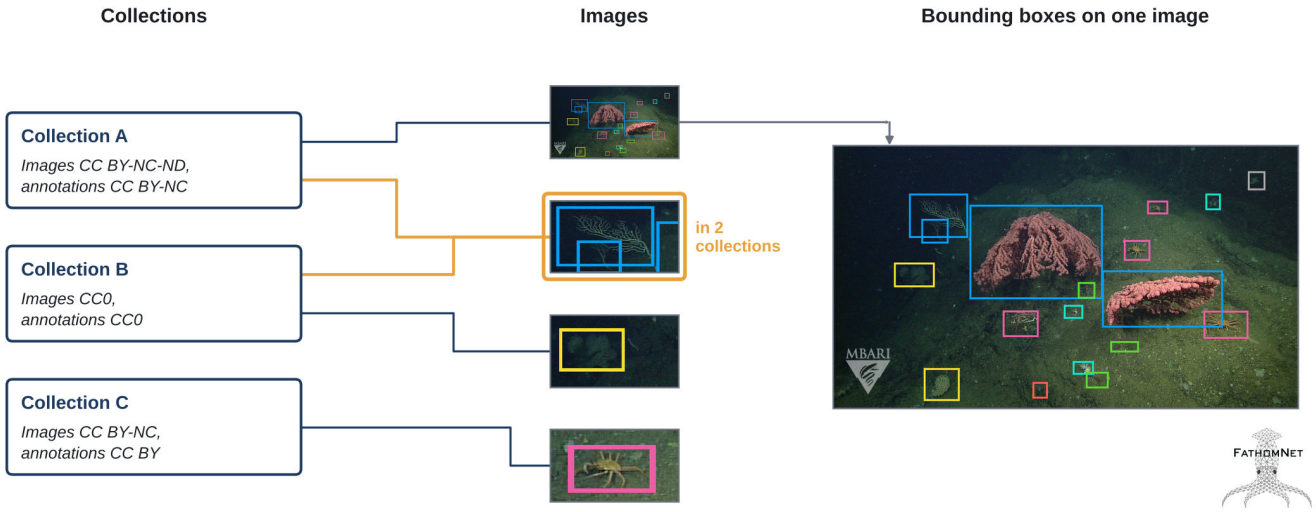


Fig. 1 | Collections, images and annotations in the FathomNet Database. Each upload batch (collection) carries one license sentence that covers both its images and their annotations; the three image and annotation license pairs shown are real combinations parsed from those sentences. An image can belong to more than one collection (amber), so its terms can come from two sentences that disagree; 5,378 images belong to two or more collections, and we did not count how many of their sentences disagree. Each image in turn carries many bounding boxes (right). The layout is schematic: collection assignments are illustrative, the thumbnails are crops of one frame from the FathomNet Database web assets (image: MBARI), and box colors encode nothing.

The contribution is the unit of license

We use contribution in the sense of the Creative Commons 4.0 licenses[5], which grant rights in Licensed Material and let a person who adapts it license the result under terms of their own (an Adapter's License). A contribution is one such piece of work as the database records it: an image, a box drawn on that image and a mask derived from the box are three contributions, possibly by three parties. This is a working definition for schema design and not a legal finding, since whether a given annotation is protected by copyright varies by jurisdiction. Ownership is a separate matter: under the FathomNet Terms of Use, annotations created or modified in the Database itself belong to MBARI[6], so licenses held by contributors apply to annotations uploaded with their imagery. A schema follows the model by making the contribution a first class row, distinct from the image it describes and from the shape it takes. Four design principles follow (Table 1).

Table 1 | Four design principles of contribution level licensing.

	Principle
P1	**Separate axes.** Images and annotations are licensed separately, with one license column per party that can hold rights independently (an inherited default is not a party).
P2	**Contribution, not collection.** A collection is closed when created, whereas contribution is open ended, so later work would otherwise be described by terms chosen without knowledge of it.
P3	**Contribution, not representation.** The license is stored once per contribution, in a record separate from its shape, so a license has exactly one home.
P4	**Shared facts, independent claims.** Facts about the subject are shared; claims by contributors are independent.

P2 and P3 carry most of the argument. Licensing in many community corpora is recorded per collection or per image and rarely per annotation, which is convenient at ingest and wrong afterwards: a per image license solves the image half and leaves the annotation half untouched, because annotations arrive years later from parties the image record knows nothing about. Violating P3 incurs technical debt. A corpus that supports several annotation shapes will naturally give each its own table, and attaching the license to those tables looks like it satisfies P2. It fails three ways: one legal value lives in several physical columns with no constraint relating them; no single query can ask what license a record carries without first establishing its

shape; and the cost of a new shape grows with the number of attributes that cut across shapes. Fixing P1 or P2 is a column addition plus a backfill; fixing P3 is a table split that gets harder with every record and every shape, which is why we chose to make the split while the database held a single shape.

In the model as built on the replica (Fig. 2), licenses are stored as one of seven Creative Commons identifiers in columns guarded by a CHECK constraint, with image terms and annotation terms in separate columns (P1). For comparison, the FathomNet Terms of Use offer four licenses for imagery and three for annotations[6], and the maintainers' v2.6.0 uses an application level list of 77 SPDX identifiers (21 selectable for images and 19 for annotations) with no CHECK constraint. In the proposed schema every new upload must carry a license, either a stated one or the image's annotation license default. The columns accept NULL so that the one time translation of legacy prose can leave rows it cannot resolve for a person instead of guessing (upstream v2.6.0 instead defaults unreadable prose to CC BY-NC-ND), and there is no sentinel value. The nongeometry columns of `bounding_boxes` move to a new `localizations` parent that holds one row per contribution, and `bounding_boxes` keeps only geometry (P3). Points and run length masks are new sibling tables on the same pattern; the FathomNet Database holds no points or masks, so these tables were exercised only by tests. Identifiers come from the parent's allocator so that a point and a box can never collide, and a mask's extent box is computed on the server from the decoded run length data. The work comprises the seven migrations of the series as adopted after review (Extended Data Table 1; three more, V1.9.0, V1.11.0 and V1.12.0, were dropped or held) on a 50 commit backend branch (`fathomnet-rest-api`), from which 17 pull requests were opened upstream (#37 and #40 to #55), with companion branches of 108 commits on the web frontend (`fathomnet-web-ui`) and 7 on the Python client (`fathomnet-py`). It deletes no upstream file and removes no endpoint or feature. The backend branch adds a `/licenses` endpoint, CRUD for points and masks and a `/localizations` surface for reads across types, all governed by one static access rule. Because Flyway will not apply a version below one already applied to a database, migration order fixes pull request merge order (Extended Data Fig. 2). Of these pull requests, only the security repair described below was merged as written; the maintainers ported the content of #44 and #46 into v2.6.0, and the outcome for the rest, including the simpler variant of P4 the FathomNet team adopted, is recorded in the Discussion.

Image and annotation terms differ on half the database

If the two license columns of P1 held the same value on every image, the second would be for show. Across the 472,639 images of production as copied on 28 July 2026, 240,952 (51%) carry an image license that differs from their annotations' license (Fig. 3): the 240,912 no derivatives images, whose annotations are CC BY-NC, plus 40 others. An image's annotation license here is the license its boxes share; no image in a live sample of 1,000 had boxes under two licenses. The live database in October 2026 gives the same 240,952 differing images. The result shows that images and annotations need separate license values (P1); it does not by itself show independent rights holders, because nearly all of it is one pattern chosen on a single upload form. With one license column, the data would force a choice between two bad outcomes: it would either extend a no derivatives restriction to the annotations on a quarter of a million images, blocking uses their authors permitted, or drop the restriction from imagery whose owners asked for it.

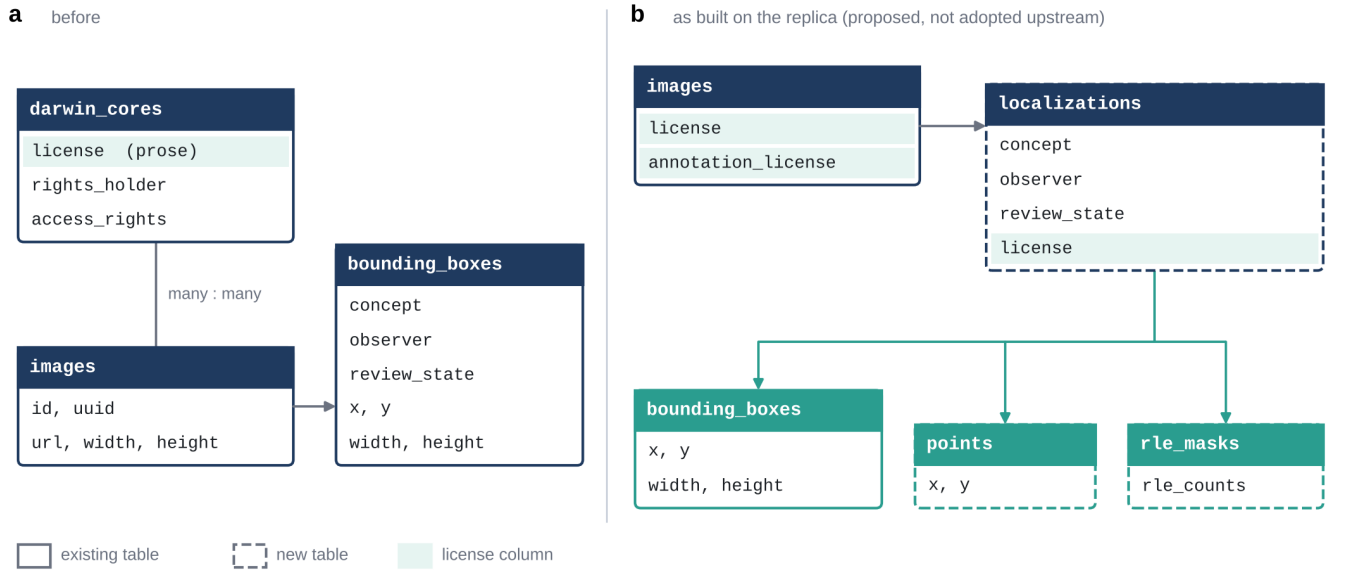


Fig. 2 | The schema before and as built (proposed; not adopted upstream). a, Before: one prose license per upload batch, and a bounding box row that carries claim, claimant, review state and shape together. **b**, As built on the replica (proposed in the closed pull request #45; not adopted upstream): two license columns on images (P1), a `localizations` parent that holds one row per contribution and its license, and geometry tables keyed to the parent (P3). The FathomNet Database is moving to a single table localization model instead (Extended Data Fig. 6b); this design was exercised only on a pseudonymized replica. Dashed outlines are tables added in this work and shaded rows are license columns. Audit tables and unchanged tables are omitted.

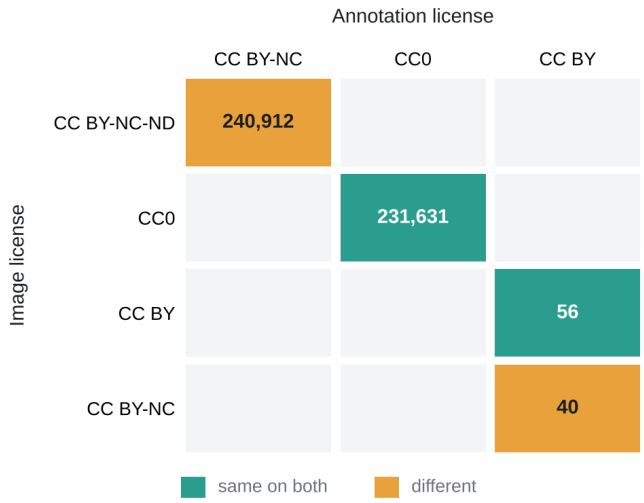


Fig. 3 | Image license against annotation license. Images per cell. Teal cells hold images whose two licenses are the same; amber cells hold images whose two licenses differ, 240,952 of the 472,639 images in production as copied on 28 July 2026 (51.0%). Gray cells are combinations that do not occur. Development test images added later are excluded.

The structure is intact at the series tip (Extended Data Table 2): each of the 1,284,154 bounding boxes is one contribution in the `localizations` parent, although we have not run an anti join to confirm that no contribution lacks a shape and no shape lacks a contribution.

One qualification matters: the contributed corpus is bounding boxes only. On the replica as copied, after our backfill, boxes occur under three licenses (CC BY-NC on 643,973, CC0 on 640,070 and CC BY on 96; Extended Data Fig. 1b); these counts have not yet been compared with the box licenses production has carried since v2.6.0. Per contribution licensing is therefore demonstrated end to end for boxes and mechanically for points and masks. The design nonetheless makes expressible what the prior schema could not: one observation described three ways, by three parties under three licenses, is three rows of the `localizations` parent, each with its own geometry row (Extended Data Table 3, a test fixture).

The model holds on a full replica of production

Each destructive migration is written to verify before it drops. To test the guard and rollback mechanism, we induced a guard failure on the full replica, on V1.9.0, an intermediate migration that was later dropped (Discussion): all 40 of its batches rolled back, and the row counts and schema matched the premigration state. No failure was induced on V1.3.0, the migration that drops columns from `bounding_boxes`. The migrations then ran twice on 29 July 2026 against a pseudonymized replica of production (30,032,224 rows, 84% of them in two audit tables; Extended Data Fig. 1a), once in a container and once on a Windows Server host with Flyway running at application start, which is the path production takes. Every license count we had fixed in advance (commit `9cf49b9`) reproduced exactly on both runs, and the quarantine table, which receives any sentence the backfill cannot parse, came back empty. In the run reported, all 574 backend tests across 75 classes passed at the tip of the stack; two review tests depend on an unseeded random fixture and fail on about one draw in ten (Methods).

Separately from the schema work, one change has already landed in the upstream backend. Quarkus 3 processes only the Jakarta spelling of the security annotations, but `BoundingBoxResource` imported the `javax` spelling, so its `@RolesAllowed` and `@PermitAll` annotations were inert on all five write endpoints (create, delete, bulk review, bulk rename and CSV upload); three of them, create, delete and bulk rename, had no other check and accepted unauthenticated requests. Pull request #37[7], merged on 31 July 2026, switches the import and adds a test that scans the source so that the `javax` spelling cannot return, and that probes the endpoints with and without credentials so that the role contract is pinned and not merely the presence of authentication. An authenticated contributor who holds only the default read role now receives a 403 where the request previously succeeded; the test suite pins this, and the maintainers report the fix deployed in release v2.4.1 on 1 August 2026. The role sets were left exactly as declared, and widening them is a separate decision. The maintainers followed up the next day (#38) with a Maven Enforcer rule that bans most `javax` dependencies from the build.

A proof of concept run derives masks from the boxes

With points and masks storable in the schema as built on the replica (upstream status in the Discussion), the next question was how they could be filled, and as a proof of concept we derived masks from the existing boxes. Boxes were pulled through the FathomNet API, written as sharded COCO[8] and prompted into the SAM 2.1 image predictor[4] (Fig. 4a). A taxonomy filter against WoRMS[9], which keeps the 2,712 of 2,858 concepts whose ancestry reaches the *Biota* root, was meant to restrict the run to organisms but was added after it: the delivered masks cover 2,855 concepts, including 5,210 review tier masks on 135 non organism concepts such as sampling gear. Each mask is graded into a composite of predicted IoU, stability, box containment, cleaning delta and polygon fidelity, and independent veto checks cap the tier whatever the score: hard rejects for masks that are wrong rather than uncertain, and review caps for masks that are plausible but need a person. An alignment guard routes any image whose returned mask count differs from its box count to review instead of reassigning masks by position.

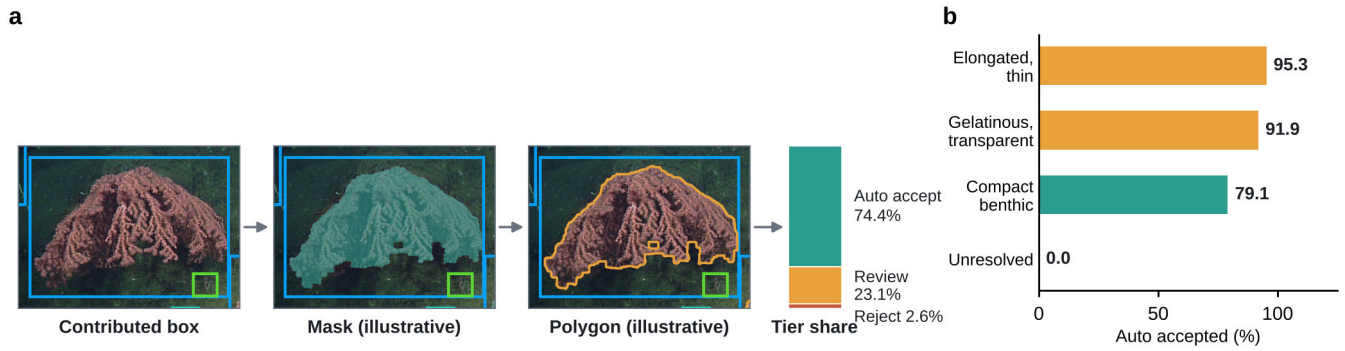


Fig. 4 | From box to mask (schematic on a real frame). a, A contributed box on a FathomNet Database frame (image: MBARI) is prompted into SAM 2, cleaned into a polygon and graded into a tier. Only the large box is shown. The mask and polygon are illustrative, made by color thresholding for this figure, and are not SAM 2 output. The bar gives the tier shares of all 1,169,279 masks in the proof of concept run. **b**, Auto accept rate by morphology group. Amber marks the two groups expected to be hardest, where a high rate needs checking against a reference set; unresolved concepts are capped at review by design.

The run completed on 20 July 2026 across all 409 shards. Of 1,169,279 graded masks, 869,377 were auto accepted (74.4%), 269,597 went to review (23.1%) and 30,305 were rejected (2.6%); 2,070 of the 1,171,349 boxes requested (0.18%) were never downloadable. Auto accept tracks morphology closely, although each group has its own thresholds, so the rates are not a direct measure of difficulty: 95.3% for elongated and thin taxa, 91.9% for gelatinous and transparent, 79.1% for compact benthos and zero for the unresolved group (Fig. 4b).

The two highest rates belong to the two groups expected to be hardest to segment, and that is a reason for caution and not a success. A transparent bell or a thin arm gives the model little contrast to follow, and the failures it can produce, a mask that fills the box or one that skips the faint parts, still score well on predicted IoU and box containment, which the composite rewards. The SAM 3 comparison below found such failures among SAM 2's gelatinous masks; it found no such pattern for elongated taxa, where SAM 2 was judged the better model, and it is too small to estimate the failure rate among accepted masks. Until a reference set exists, auto accept precision is unvalidated: the true precision of accepted masks could be lower, and some review tier masks may be correct.

The finding that mattered most was calibration, not modeling. The seed concept map covered 31 of 2,858 concepts, because it had been written against competition common names while the database carries scientific ones, so all but those 31 concepts fell to the review capped default group and the first real run auto accepted nothing at all. Resolving WoRMS ancestry per concept lifted coverage to 2,200 of 2,858 and took a 320 box pilot from zero to 86.2% auto accepted (Extended Data Fig. 3). Because score and vetoes are stored per mask, thresholds can be recut without rerunning SAM 2, and once a reference set exists, per taxon precision can be certified with a lower confidence bound of the Clopper–Pearson type[10].

Recording licenses per image in the proposed schema makes a further question answerable: which masks could be ingested or shared. Applying a proposed license check (`DerivationPolicy`, from the closed pull request #50 and not implemented upstream) would hold back masks on the 240,912 images (51%) under no derivatives image terms; we did not count how many of the 1,169,279 masks fall on those images. Creative Commons no derivatives terms permit making adapted material but not sharing it[5], so the masks could be produced and only their ingest and redistribution are restricted. The rule reads the image license only. No annotation in the database carries no derivatives terms (Fig. 3), so checking the box's annotation license would exclude nothing more today, and a pixel derived mask is treated more strictly than the boxes on the same images. This is a cautious choice and not a legal finding. None of the masks has been ingested.

Sharded execution kept one GPU busy

The proof of concept run is a sharded job coordinated by atomic transactional leases (Fig. 5). Boxes are written at 1,000 images per shard, 409 shards in all, and a SQLite manifest coordinates the workers: a worker claims a shard with an atomic lease, the lease of a dead worker is reclaimed so that its shard returns to the queue, and each shard's output is written to a temporary file and atomically renamed into place, so that a shard is either absent or complete (Extended Data Fig. 4). Scaling was limited by starvation, disk and memory until the GPU was saturated, and four bottlenecks were found and fixed in turn.

First, a single worker left the A100 about 70% idle while it alternated GPU encoding with CPU polygon and grading work; four workers sharing one manifest raised GPU use from about 30% to roughly 95 to 100% and roughly tripled GPU utilization. Second, serial downloading in front of the model, at about 1.2 images per second, projected a run of about 90 hours; a concurrent prefetch that downloads while the GPU masks took downloading off the critical path. Third, with every worker prefetching a whole shard, the image cache grew to 18 GB on a 32 GB disk and filled it at shard index 290 of 409, failing 16 shards on write; capping the prefetch at 48 unconsumed images and evicting each image once decoded held the cache near 600 MB however long the run. Fourth, one shard took about 30 minutes with the GPU idle and no out of memory error: on dense frames the decoder's tensors grew with the box count, and the allocator fragmented up to the 40 GB ceiling and thrashed. Chunking prompts 16 at a time, emptying the cache per image and switching to expandable allocator segments held each worker near 2 GB, with four workers at about 95% GPU use.

Two of the four bottlenecks were starvation and one was disk capacity, so a faster GPU would not have helped with them, and the fourth was fragmentation amplified by the demand of dense frames. The disk full failure was recovered by stopping the workers, clearing the cache, resetting failed and claimed shards to pending and redeploying; no completed shard was lost, because completion is an atomic rename.

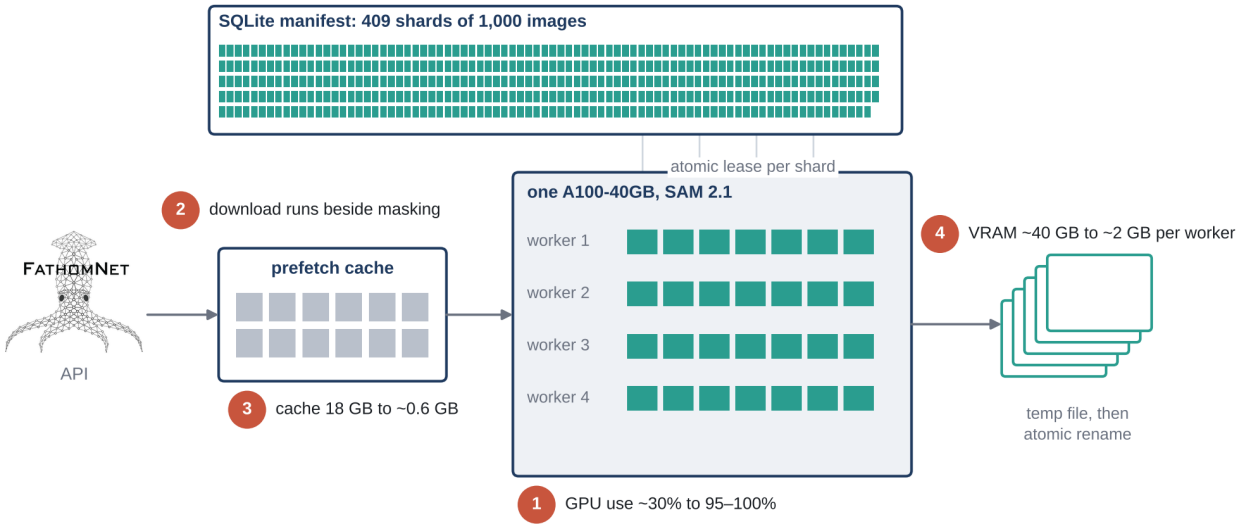


Fig. 5 | How the shards ran. Workers lease 1,000 image shards from a SQLite manifest (top; one cell per shard, all 409 completed on 20 July 2026), mask prefetched images on one A100 and write each shard by temporary file and atomic rename. Red markers place the four bottlenecks and their fixes: 1, four workers on one manifest (GPU use from about 30% to roughly 95 to 100%); 2, downloading overlapped with masking (serial download had projected about 90 hours); 3, a bounded prefetch with eviction (cache from 18 GB to about 0.6 GB); 4, prompts chunked 16 at a time with an expandable allocator (VRAM from the 40 GB ceiling to about 2 GB per worker). Blocks inside the boxes are schematic.

SAM 3 matches SAM 2 on the median and may split by animal shape

SAM 2 was run first because the pipeline had been built around it. SAM 3[11] was then brought to our attention for its stronger image encoder, and because both runs were done, this section is a small proof of concept comparison of SAM 2 with SAM 3, whose encoder, among other components, differs. SAM 3 is also familiar with this kind of imagery: FathomNet is one of the subsets of its SA-Co/Silver evaluation benchmark[12], so it was evaluated on FathomNet images during development, although we found no record of whether FathomNet images were also used in training. SAM 2 encodes the image with Hiera, a hierarchical encoder whose high resolution early stages pass fine detail to the mask decoder[4]. SAM 3 replaces it with the Perception Encoder[13], a vision transformer trained contrastively on image and text pairs at very large scale, whose features encode what an object is and not only where its edges are; that is what lets SAM 3 segment from a text concept as well as from a box[11].

A 350 mask sample, stratified over the two hardest morphology groups plus a compact control, was box prompted through SAM 3 and compared with the SAM 2 mask from the run. IoU between the two masks measures agreement and area ratio measures whether one model is systematically tighter; neither says which mask is right. On the most divergent fifth, where the models disagree, two vision model judges compared 30 panels from each of the two hard groups (120 judgments; the compact control was not judged); five of five spot checks agreed with our own reads, which is a sanity check only and is not recorded in a file. On the median the two models agree closely, at IoU around 0.8 and area ratio around 1.0. That is expected, because a box prompt already tells the model where the object is, which removes most of the advantage of an encoder that is better at recognizing it.

The models separate only on the most divergent fifth, and there the result appears to split by morphology (Extended Data Fig. 5a). On gelatinous and transparent panels SAM 3 was judged better on 16 of 30 (0.53; 95% Clopper–Pearson interval 0.34 to 0.72) and SAM 2 on 8, with one tie and five panels where the judges disagreed; on decided panels alone, SAM 3 won 16 of 24 (0.67). In those panels SAM 3 rescued masks that filled the box and recovering bells and tentacles that SAM 2 missed; a plausible reason, which we have not tested directly, is that semantic features recognize a faint, low contrast body as part of the animal where features driven by edges have little to follow. On elongated and thin panels SAM 3 was judged better on only 10 of 30 (0.33; 0.17 to 0.53) and SAM 2 on 15, with three ties and two disagreements (10 of 25 decided, 0.40); there SAM 3 misses fan arms, grabs background or returns an empty mask; a plausible reason is that the transformer works on coarser patches than Hiera's early stages, so structures a few pixels wide fall below its resolution. Both intervals are wide, so the split is suggestive and not established. If the split holds against a reference set, the useful outcome would be a targeted rerun of gelatinous and transparent masks already flagged as failures, with SAM 2 kept for elongated taxa. For a decision of that kind the metrics that matter are IoU and boundary accuracy against a reference set, together with the empty mask rate, and the judged win rate stands in for them until a reference set exists. Two other avenues were tried and dropped on informal observation, without counts. HQ-SAM tightens masks by roughly 20% uniformly, but tighter was never shown to be better against any reference. Open vocabulary gear negatives fail because Grounding-DINO reads crab legs, echinoderm arms and thin organisms as cable and machinery and flags the animals themselves.

The Perception Encoder of SAM 3 was also run once over the whole database, storing one 1,024 dimensional embedding per unique image, pooled over the backbone, for clustering, deduplication and novelty search. It covered 407,161 of 408,860 unique images (99.6%) in about 21 hours on one machine and was limited by download speed throughout (Extended Data Fig. 5b; Methods).

Discussion

Recording a license at the level of the contribution makes terms queryable and correct at export, and the measurement that justifies it is simple: the image license and the annotation license differ on half the FathomNet Database. The same terms decide which machine derived annotations the database can take in. A proof of concept run can propose over a million masks and sort them by trust, and the

proposed license columns would say which of them may be ingested or shared.

The FathomNet team settled on a simpler model than the one built here, and both belong in the record (Extended Data Fig. 6). As built, a license is a claim by a contributor about their own work and so should not sit on a row shared by contributors: a person draws a box under CC BY-NC and a second annotator later traces their own outline of the same animal, which gives one animal, two contributions, two contributors and no reason for their terms to agree. A further migration (V1.9.0) therefore raised an `annotations` parent above `localizations` to hold facts about the subject. The model the team chose in review, as a group decision (a per annotation license and a has-a relation), rests on two grounds. First, the license is per annotation: one annotation, with all of its geometries, keeps the license it was created with, which is a model that a contributor can hold in their head and a consumer can query without a join, and which is consistent with the license choices in the FathomNet Terms of Use[6], although the Terms do not address whether a license may change. Second, the relation between an observation and its shapes is *has-a*, not *is-a*: joined table inheritance asserts that a localization *is* a box, exclusively, whereas the database needs a localization that *has* geometries, optionally and nonexclusively. This is a change of structure as well as cardinality. In V1.3.0 each geometry type has its own child table under a localizations parent, whereas the model the team now favors (next paragraph) keeps every geometry in a single typed table keyed to the observation. The two vocabularies also differ: there, an observation is the parent and each localization is one geometry, at most one of each type; here, a localization is a contribution, and annotations were raised above it.

The maintainers have since gone further. On the upstream backend, #37 is the one pull request merged as written; the maintainers integrated fixes from #41 and #43 into release v2.5.2, and the rest were closed without merging. The granular licenses pull request (#44) and its backfill (#46) were closed after the maintainers ported their content into their own licensing pull request (#61), released as v2.6.0 on 10 September 2026. That release stores an image license on each image and an annotation license on each bounding box, which satisfies P1; it uses an application level list of SPDX identifiers rather than a CHECK constraint, and it resolves unparseable legacy prose to CC BY-NC-ND rather than queuing it for a person. The pull request that introduced the joined inheritance localizations parent (#45) was closed with a note that the team is looking at a single table approach, and the derivation provenance and per image annotation terms (#50) with a note that the team is looking at aligning with a draft proposal for annosaurus, MBARI's annotation service[14], which would store each geometry in one typed table with at most one localization of each type per observation and does not yet address licenses. The remaining pull requests (#47 to #49 and #51 to #55) were built on #45 and #50 in a single chain of commits and were closed unmerged. This route can still satisfy P3, provided that the license is stored on the observation that the typed localization rows are keyed to, and not on the localization rows themselves: an observation can carry a box, a point and a mask, each with its own provenance columns. P3 still has to be carried into the single table design.

In the model chosen in review, a model derived mask joins the box's annotation and takes its license, and for the mask import that is the right default. A mask derived from a box is Adapted Material in the Creative Commons 4.0 sense, and its terms must not prevent recipients from complying with the box's conditions[5], so inheritance is the conservative choice. The case P4 protects is different: independent work by a second contributor on an animal that someone else has already annotated, who would need a separate annotation, which asserts a separate observation of the same animal. 'Keeps the license it was created with' also means that a license cannot change after creation, and nothing enforces that today: in v2.6.0 both the image and the bounding box update endpoints accept a changed license. In the local stack the change is contained but not free. V1.3.0's geometry tables share the localization's primary key, one shape per contribution; the chosen model lets one localization carry several geometries, so the geometry tables would need their own keys and a foreign key to the localization, and the one shape integrity check would become at least one shape per localization. V1.9.0 and the derived geometry flag of V1.11.0 are dropped, and V1.12.0, an image encodings table with no producer and no consumer in any repository, is held. Dropping V1.9.0 removes the largest and least evidenced migration in the stack: 469 lines over 40 batches that reshape the audit trail a second time across 1.28 million rows, never timed and shipping no rollback script of its own, although its in migration guard did roll back cleanly when tested. Upstream the question is moot, because the maintainers are moving to a single table. The migrations that implement P1 to P3 are unchanged and the 51% result is unaffected; for model derived geometry, the chosen model gives later work the terms of the earlier box, which is the case P2 guards against, accepted here for simplicity and supported by the license terms themselves.

P4 remains a reasonable claim about annotation corpora in general, because the adopted model chooses simplicity for the FathomNet Database and does not find a fault in the argument. A principle derived from how rights work met an institution with published terms of use, a small maintainer team and good reasons to prefer a model that users can hold in their heads. The case P4 protects is named; supporting it would need a per contribution license on whatever single table design is adopted, and the mask import is not that case.

Our results have limitations. None of the seven migrations was merged as written, so the schema described here has been exercised only on a local replica; the upstream repair (#37) is the one pull request merged as written, and the license columns and backfill of #44 and #46 were ported into the maintainers' v2.6.0, which differs from ours as described above. The derivation policy and the per image annotation terms were part of the closed #50, so the license check described above is a proposal for the FathomNet Database's ingest path and not yet part of it. The contributed corpus is bounding boxes only, so points and masks are demonstrated mechanically and not on real contributions. Auto accept is a tuned threshold, not checked against a reference set, and the predicted IoU that enters the grade also chose the best of three candidates, so the grade is partly circular. The delivered masks were not filtered to organisms. The SAM 3 comparison is small and model judged. Finally, recording is not enforcement: the work makes terms queryable and correct at export but does not constrain what a consumer does with exported data.

Several open questions are policy and not engineering. Every COCO dataset exported by the Python client, in every release through the current v1.10.2, carries a license block that names a license that does not exist, at a URL that returns a 404 page, and there is no way to find those datasets; whether to issue a correction notice is undecided. Third party datasets need a per dataset license decision made once before any import adapter runs, and it is unsettled whether publishing an inference time mask counts as training and development under the machine learning override in the FathomNet Terms of Use[6], which covers visual content and does not mention annotations. And one change breaks an external client: once the annotation license default ships, creating an image without an annotation license

returns 400 where it returned 200, so JSON ingest clients must be updated before that deploy.

Future work follows directly. The first task is to carry an immutable per annotation license into the single table design; v2.6.0 already keeps image and annotation licenses separate (P1). The immutability rule needs a constraint, trigger or endpoint that refuses, before the database has a reason to test it. The FathomNet Database has no reliable field that records whether an annotation was drawn by a person or produced by a model (the free text observer is overwritten on rename, and `basisOfRecord` is per dataset; the annosaurus proposal would add one[14]), and that gets harder to answer with every model generated row. A reference set would let per taxon mask precision be certified and would justify the targeted SAM 3 rerun. We hope that the four design principles are useful to other annotation corpora before their schemas are fixed, while the costly repair (P3) is still a design choice.

Reflection

The main lesson is about sequencing and not about the design. The backend maintainers were involved at the final review of everything, and in hindsight that was too late: the design needed their review early and at every step, not once at the end. Feedback from the FathomNet team arrived in rounds (the has-a decision took three), and a stack of thirteen dependent pull requests made every round costly, because a change at the bottom invalidated everything above it. A short proposal, read by the people who own the backend before the first migration was written, would have surfaced the disagreement over a joined inheritance parent as a comment on a document, before a stack of pull requests depended on it.

Two things held up. The one pull request merged as written, #37, was small, self contained and tested, stated its blast radius and left the role decisions to the maintainers; that is the pattern that works. And the measurement and the replica verification gave the discussion facts and not opinions. The principles do not depend on table layout: a single table can satisfy P3 if the license has one home, and the maintainers' v2.6.0 already keeps image and annotation licenses separate (P1).

Next time we would open with the proposal and the measurement, ask for a decision on the physical design before writing any migration, land the smallest independently useful step first, and keep the maintainers in the review loop on every step.

Methods

Production replica and counts

All counts come from a pseudonymized replica of production copied on 28 July 2026, which held 472,639 images, 1,284,139 bounding boxes and 346 Darwin Core rows. Counts of the replica as loaded, before any migration, are 30,032,224 rows across 16 tables, because Hibernate Envers keeps a parallel audit table per audited entity; `images_AUD` (13,009,122) and `REVINFO` (12,288,651) account for 25.3 million, and the live table that the riskiest migration rewrites, `bounding_boxes`, held 1,284,139 rows. Migrations that touch audited entities therefore run over tens of millions of rows. Counts at the series tip (Extended Data Table 2) were measured after all migrations in one query batch on 9 August 2026; development and test sessions had by then added 14 images (all CC BY on both axes), 15 boxes (six CC BY-NC, four CC0, three CC BY and two unlicensed fixtures), giving 472,653 images and 1,284,154 boxes. The collection counts (346 Darwin Core rows, 245 collections sharing one string, 212,795 images under it, and 5,378 images in two or more collections) come from production as copied; 5,378 is the excess of collection memberships over images, and the live database gives the same figure. Since 245 rows share one string, the 346 rows hold at most 102 distinct strings; the live database in October 2026 holds five across 350 uploads. The license cross tabulation (Fig. 3) sums as 240,912 + 231,631 + 56 + 40 = 472,639 images. In the live database in October 2026 the image license filter returns the same 240,912 CC BY-NC-ND and 231,631 CC0 images.

Schema and migrations

In the proposed schema, licenses are `varchar(64)` columns with a CHECK constraint over seven identifiers, including CC0, CC BY, CC BY-NC and CC BY-NC-ND. In the proposed schema new uploads must carry a license; NULL remains only on legacy rows the backfill did not resolve (on the replica, two development test boxes with no license prose), and there is no sentinel value. Migration V1.3.0 verifies and then destroys: ten THROW guards inside one transaction check counts, membership and audit shape before any column is dropped (36.8 s on the 1,284,139 rows of the replica as loaded). V1.4.0 backfills only empty rows from the prose and queues conflicting or unparseable prose in a quarantine table for a person. Each migration commits its own transaction, so a data dependent failure in the backfill leaves the table split intact and a retry reruns only the backfill. The lineage pointer `derived_from_uuid` has no foreign key, because SQL Server forbids ON DELETE CASCADE or SET NULL on a self reference (error 1785). `images.annotation_license` is materialized (V1.8.0) so that the annotation create path has a defensible default in one indexed lookup. A license constrained retrieval needs no knowledge of shape: `SELECT uuid, concept FROM localizations WHERE license = 'CC0'` returns correct results today and will keep doing so when a fourth shape is added as one table with a shared primary key.

Verification

The migrations ran twice on 29 July 2026 against the replica, once in a container and once on a Windows Server host with Flyway running at application start. Every license count we had fixed in advance (commit `9cf49b9`: 643,973 CC BY-NC, 640,070 CC0 and 96 CC BY boxes) reproduced exactly on both runs, and the quarantine table came back empty. A guard failure was induced deliberately on V1.9.0, the largest migration of the series (the annotations parent, later dropped): all 40 batches rolled back. No failure was induced on V1.3.0. The backend suite has 574 tests across 75 classes. Separately, nine new Python client tests were run against the unmodified source and failed six ways, so they detect the defects they target. One caveat applies: the test fixture marks roughly one box in ten as already verified from an unseeded random source, and two review tests fail on the draws that hand back such a box, so a passing count describes one draw and not a property of the branch.

Upstream access control repair

Pull request #37 changes five files (+149, −5 lines; squash commit `e955f78`) and was merged on 31 July 2026. It switches the `BoundingBoxResource` imports to the Jakarta spelling and adds `ResourceSecurityAnnotationTest`, which scans the source for the `javax` spelling, probes the endpoints that lack an in method authorization fallback without credentials, and probes with a read only identity. An existing upload test had authenticated as a random subject with no roles, which was invisible while the annotation was inert; it now authenticates with the declared role. Merge details are from the upstream pull request record; the maintainers report the fix deployed in release v2.4.1 on 1 August 2026.

Mask pipeline

Boxes were pulled through the FathomNet API with offset pagination and up to eight retries with backoff on HTTP 429, 500, 502, 503 and 504 and on connection errors (no request timeout was set), and written as sharded COCO at 1,000 images per shard with a resumable SQLite checkpoint that provides stable ids and UUID deduplication. A taxonomy filter that drops any concept whose WoRMS ancestry lacks the *Biota* root, which removes gear, substrate and detritus and leaves 2,712 of 2,858 concepts, was added after the run; the delivered masks span 2,855 concepts, including 5,210 review tier masks on 135 of the 146 excluded concepts. Masks came from the SAM 2.1 image predictor (hiera-large) on an A100-40GB: boxes were grouped by image, the image was encoded once, each box was prompted with multimask output, and the best of three candidates was chosen by the model's own predicted IoU.

Cleaning, scoring and veto checks

Every concept was assigned to one of four morphology groups by resolving its full WoRMS ancestry and matching the deepest higher taxon against a rule table (for example, Antipatharia overriding Hexacorallia). Each mask is scored before cleaning and graded against the emitted polygon afterwards, into a composite of predicted IoU, stability, box containment, cleaning delta and polygon fidelity. Independent veto checks cap the tier regardless of the

composite. A concept that never resolved to a morphology group has no calibrated cleaning rule, so it is capped at review however confident the model was. Review is a human queue and not a failure class.

Distributed execution

A SQLite manifest coordinates the workers through atomic transactional leases. A shard moves from pending to claimed when a worker takes an atomic lease, from claimed to done when its output is renamed into place, and from claimed to failed on error. A claimed shard whose lease has expired is reclaimed to pending automatically; failed and claimed shards can also be reset to pending by hand. Output is written to a temporary file and renamed, so a half written shard is never visible. Four workers shared one A100-40GB. Images were prefetched concurrently with masking, with at most 48 unconsumed images held and each image evicted once decoded. Prompts were sent to the decoder 16 at a time, the CUDA cache was emptied after each image, and the PyTorch allocator ran with expandable segments, which held each worker near 2 GB of GPU memory.

SAM 3 comparison and embedding pass

The comparison sample had 350 masks, stratified over the gelatinous and transparent and elongated groups plus a compact control, and was scored for IoU and area ratio between the SAM 3 masks and the SAM 2 masks from the run. Two vision model judges each judged 30 panels from each of the two hard groups on the most divergent fifth (120 judgments); panels the judges called a tie or disagreed on are counted separately, and the counts are given in the text. Five spot checks by the author agreed with the judges, a sanity check only; they are not recorded in a file. The embedding pass used the Perception Encoder of SAM 3 to store a 1,024 dimensional, fp16, L2 normalized vector per unique image. It ran in resumable 2,000 image shards. From one client the pass averaged about 5.4 images per second, with hourly rates from 0.6 to 11.1 that fell after the first ten hours, so the GPU was idle for most of the run. Of 408,860 unique images, 407,161 were embedded and 1,699 failed and were not retried. The embeddings come from SAM 3's own encoder, so they cannot serve as an independent check on SAM 3's masks.

Data availability

FathomNet Database imagery and annotations are served at database.fathomnet.org. Every count reported here is given in the text, figures and Extended Data and comes from production as copied on 28 July 2026 unless stated otherwise; the pseudonymized production replica from which they were measured is not public. The graded masks have not been ingested into the FathomNet Database.

Code availability

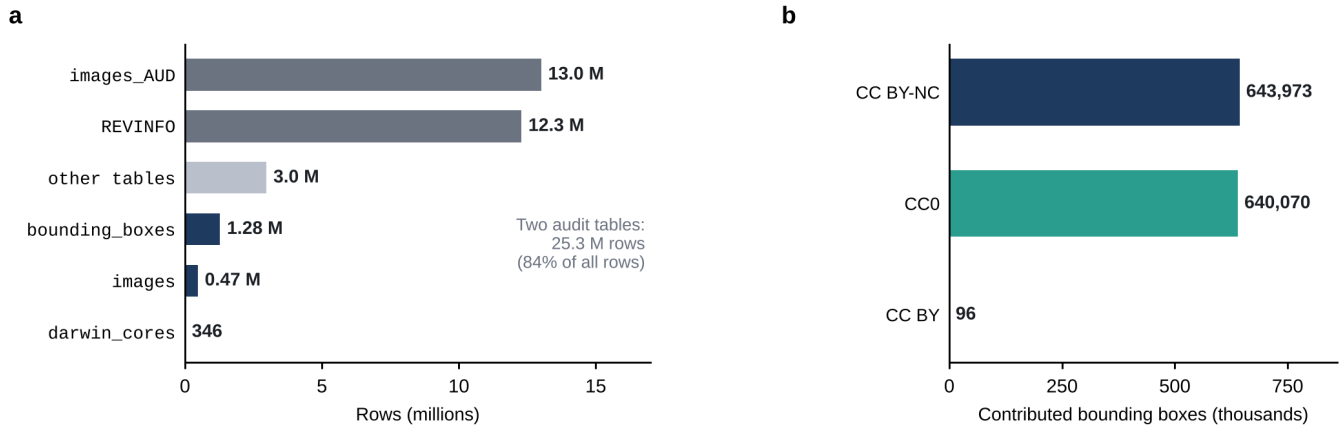
The Python client work is public at github.com/dbold23/fathomnet-py. The backend and web UI work lives in forks of the upstream repositories under github.com/dbold23; those forks are currently private, and access is available on request. The merged repair is in the upstream backend, fathomnet/fathomnet-rest-api, as pull request #37; the other pull requests (#40 to #55) were closed upstream without merging, although the maintainers ported the content of #44 and #46 into release v2.6.0 and fixes from #41 and #43 into v2.5.2; the forks hold their final state. A companion decision record (216 decisions across 25 chapters, each citing the file and, where one exists, the commit and test that carry it) is available on request.

References

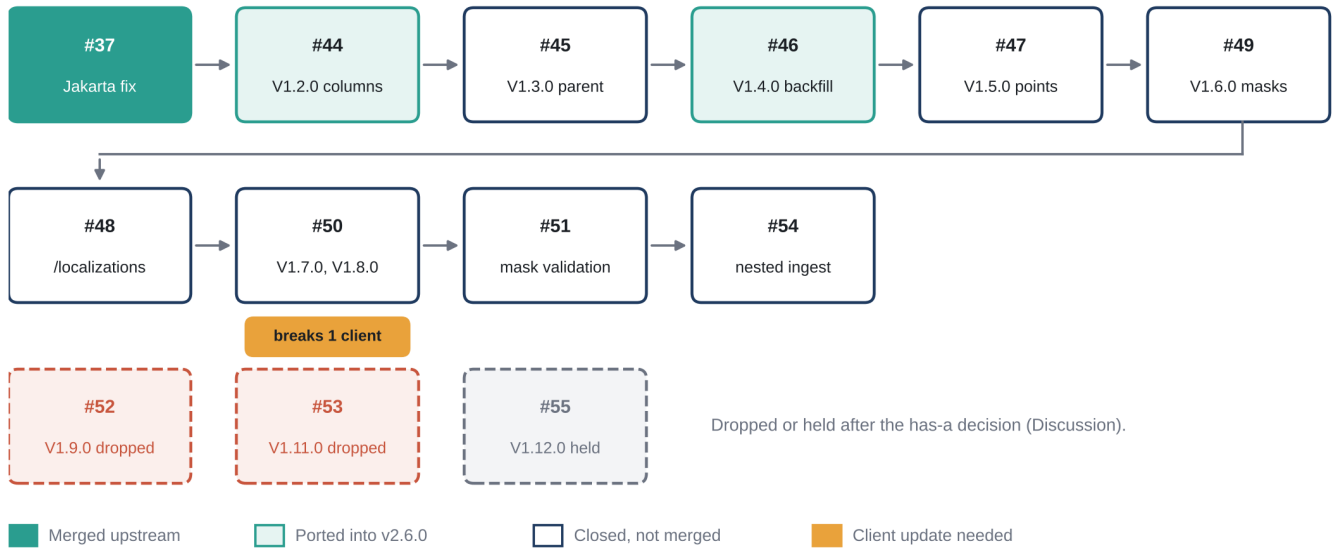
1. Wilkinson, M. D. et al. The FAIR Guiding Principles for scientific data management and stewardship. *Sci. Data* **3**, 160018 (2016).
2. Katija, K. et al. FathomNet: a global image database for enabling artificial intelligence in the ocean. *Sci. Rep.* **12**, 15914 (2022).
3. Wieczorek, J. et al. Darwin Core: an evolving community-developed biodiversity data standard. *PLoS ONE* **7**, e29715 (2012).
4. Ravi, N. et al. SAM 2: segment anything in images and videos. Preprint at arXiv:2408.00714 (2024).
5. Creative Commons. Attribution 4.0 International Public License, legal code, Section 1 (definitions of Adapted Material, Licensed Material and Licensor). creativecommons.org/licenses/by/4.0/legalcode
6. FathomNet. Terms of Use (last updated 12 January 2025) and Data Use Policy. fathomnet.org/terms; fathomnet.org/datause
7. fathomnet/fathomnet-rest-api, pull request #37: fix: use jakarta security annotations so @RolesAllowed is enforced (merged 31 July 2026). github.com/fathomnet/fathomnet-rest-api/pull/37
8. Lin, T.-Y. et al. Microsoft COCO: common objects in context. In *Proc. European Conf. Computer Vision* (2014).
9. WoRMS Editorial Board. World Register of Marine Species. marinespecies.org
10. Clopper, C. J. & Pearson, E. S. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika* **26**, 404–413 (1934).
11. Carion, N. et al. SAM 3: segment anything with concepts. Preprint at arXiv:2511.16719 (2025).
12. Meta AI. SA-Co/Silver benchmark, FathomNet subset. facebookresearch/sam3 repository, scripts/eval/silver. github.com/facebookresearch/sam3
13. Bolya, D. et al. Perception Encoder: the best visual embeddings are not at the output of the network. Preprint at arXiv:2504.13181 (2025).
14. mbari-org/annosaurus, MBARI's annotation service. Proposal: first-class localizations (draft, 25 August 2026; revised 12 October 2026). github.com/mbari-org/annosaurus

Acknowledgements

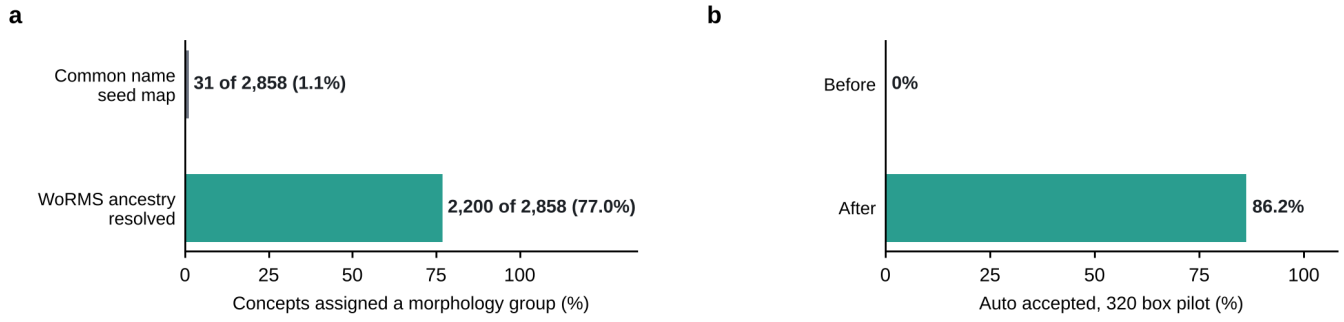
I'd like to thank Kevin Barnard, Laura Chobnik, Kakani Katija, Joost Daniels and Emily Clark at the Monterey Bay Aquarium Research Institute for the mentorship and guidance. :)



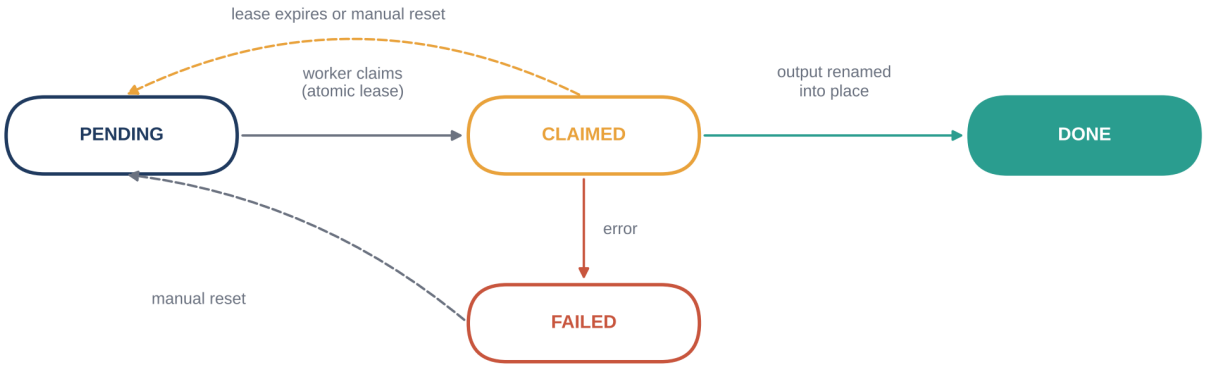
Extended Data Fig. 1 | The database at a glance. a, Row counts on the replica as copied from production on 28 July 2026: the two audit tables hold 84% of all rows, which is why every migration is costed against them. 'Other tables' is the remainder of the 30,032,224 total. **b**, License of every bounding box on the replica as copied, after our backfill; not yet compared with the box licenses production has carried since v2.6.0.



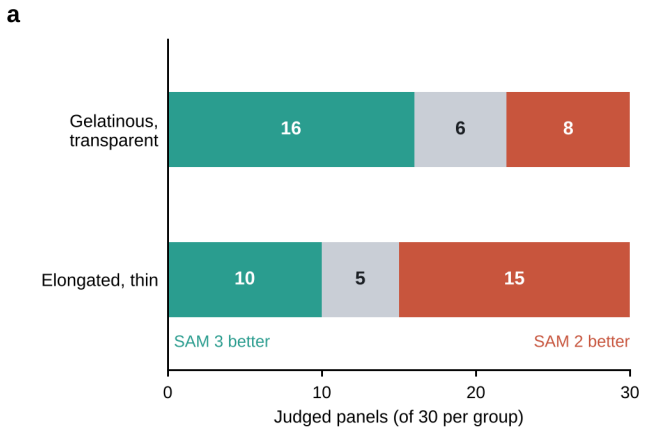
Extended Data Fig. 2 | The backend pull request stack in deploy order. Pull requests are labeled by their upstream number. #37 (the Jakarta repair) was merged upstream; #44 and #46 were closed after the maintainers ported their content into v2.6.0; the rest were closed without merging. #43, a bug fix at the base of the chain, is not shown. The amber tag marks the one change that breaks an external client. After the has-a decision, #52 and #53 were dropped and #55 held.

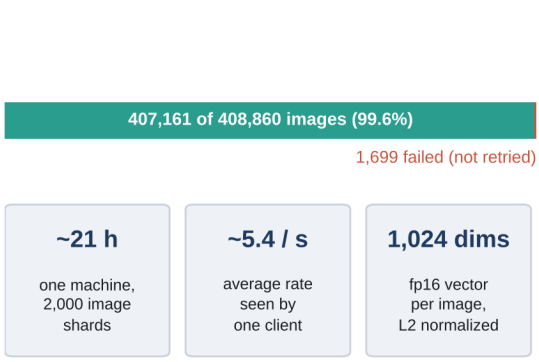


Extended Data Fig. 3 | Calibration, not modeling, decided the outcome. a, Share of the 2,858 concepts assigned a morphology group before and after resolving WoRMS ancestry. **b**, Auto accept rate on a 320 box pilot before and after.

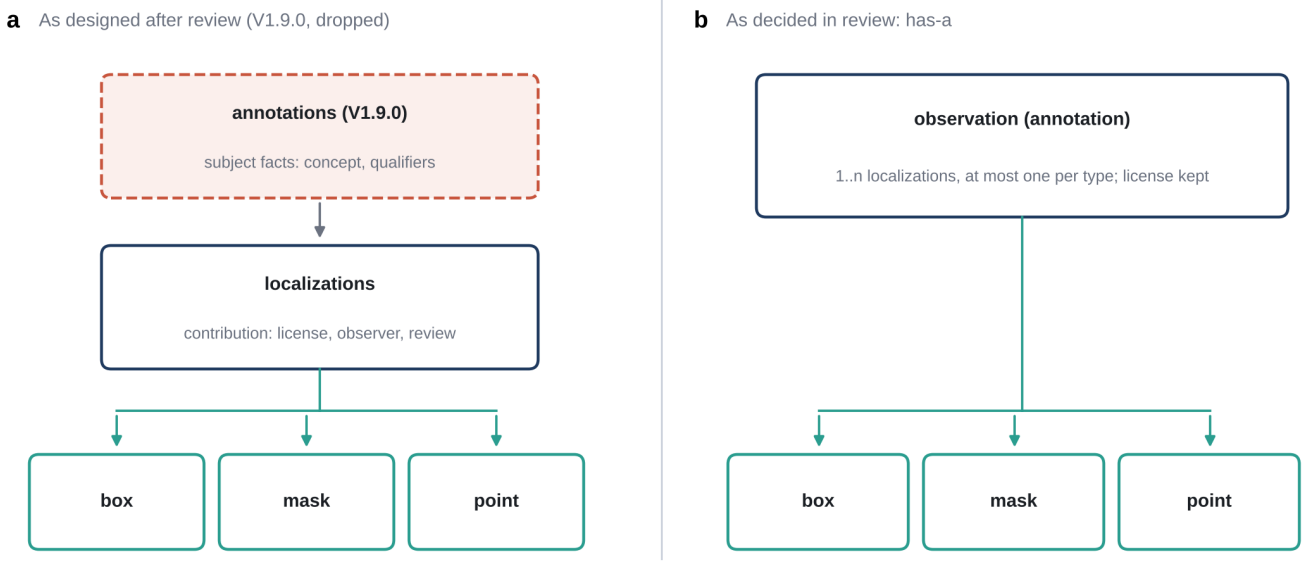


Extended Data Fig. 4 | The shard lifecycle. A shard reenters the queue in two ways: automatically, when a worker's lease expires, and by hand, when a failed or claimed shard is reset.





Extended Data Fig. 5 | The two SAM 3 runs. a, Judged outcome on 30 panels from each of the two hard groups, taken from the most divergent fifth of a 350 mask stratified sample and judged by two vision model judges; gray marks ties and panels the judges disagreed on. The compact control was not judged. **b**, The database wide Perception Encoder embedding pass, which was limited by download speed throughout; the rate is the average one client saw.



Extended Data Fig. 6 | The design after review and the model chosen in review. a, As designed after review (V1.9.0, later dropped): an annotations parent holds facts about the subject and leaves the license on the contribution. **b**, As decided in review: an observation has one or more localizations, one geometry each and at most one of each type, and the annotation keeps the license it was created with.

Extended Data Table 1 | The seven migrations of the stack, in deploy order. Version numbers are those of the fork; upstream has since used V1.2.0 and V1.3.0 for its own migrations.

Version	Change	Decision that carries it
V1.2.0	License columns on images and bounding boxes (and audit twins)	NULL only for legacy rows the backfill cannot resolve; new uploads must carry a license once the default ships.
V1.3.0	`localizations` parent; boxes reduced to geometry	Verify, then destroy: ten THROW guards inside one transaction check counts, membership and audit shape before any column is dropped.
V1.4.0	CHECK on seven values; backfill from prose; quarantine table	A refusal to guess: only empty rows are written, and conflicting or unparseable prose is queued for a person (upstream v2.6.0 defaults it to CC BY-NC-ND).
V1.5.0	`points` sibling geometry table	No sequence of its own; ids come from the parent allocator.
V1.6.0	`rle_masks` sibling geometry table	Extent box computed on the server; no client setter.
V1.7.0	`subject_uuid`, `derived_from_uuid` with two CHECKs	No foreign key on the lineage pointer: SQL Server forbids the needed referential action on a self reference.
V1.8.0	`images.annotation_license` materialized	Gives the annotation create path a defensible default in one indexed lookup.

Extended Data Table 2 | Record counts by level at the series tip (9 August 2026, including development test records).

Level	Table	Records
Image	`images`	472,653
Annotation contribution	`localizations`	1,284,154
Geometry	`bounding_boxes`	1,284,154

Extended Data Table 3 | One observation (*Bodianus pulchellus*, a test fixture) described three ways. The image is CC0 on both axes, so none of these licenses is inherited.

Contribution	Shape	License	Contributor
1	Bounding box	CC0	Legacy import
2	Point	CC BY-NC	Contributor A
3	Outline	CC BY	Contributor B