Most accounts of synthetic data stop at the dataset: you generate some images, you train on them, you report a number. For an industrial computer-vision programme that has to serve a large and heterogeneous real archive, that framing is too small, because it treats the generator as a fixed input rather than as the part of the stack whose reach determines everything downstream. This whitepaper makes the platform argument instead. We treat the procedural log generator behind our raster well-log digitiser as infrastructure: a standing capability whose coverage of three axes, image size, curve count and scan noise, is a reusable asset that a fixed model draws on to generalise across the whole archive. The archive it had to serve is 136,771 TIF scans and 7,781 LAS files; the corpus that carried it was 15,000 synthetic training curves in the version-two build, drawn from a 20,000-log two-curve generator run, with images ranging 3,200 to 12,800 pixels wide and 480 to 640 tall, evaluated across 5 loss functions, and it contained zero human annotations. The claim we defend is that this is not a dataset you rebuild per job. It is a capability whose coverage sets an upper bound on the reach of any fixed model trained on it, whose one-time build cost amortises across every real scan the model ever serves, and whose breadth, not the count of real labels, is what carried the digitiser to a peak reconstruction R-squared of 0.9891. We separate this from the two questions it is often confused with, the internal design of the generator and the architecture of the model, and argue the platform view directly: coverage is the asset, reuse is the economics, and coverage-driven generalisation is the mechanism. The deliverable is a way of budgeting, staffing and defending a synthetic-data capability as durable infrastructure that any team standing up industrial CV under a no-labels constraint can reuse.

Tannistha Maiti
Senior AI Researcher

Quamer Nasim
ML Research Engineer

Tarry Singh
Founder & CEO

Narendra Patwardhan
Research Collaborator

VEERNET / SYNTHETIC-DATA / PROCEDURAL-GENERATION / DOMAIN-RANDOMISATION / INDUSTRIAL-COMPUTER-VISION / INFRASTRUCTURE

CONTENTS

> "
>
> A generator you can dial wide enough is not a dataset you build once and throw away. It is the part of the stack that decides how much of a real archive one fixed model can serve, and that makes it infrastructure.
>
> "

I

THE REFRAMING

A generator is a capability, not a delivery

What we are actually claiming

There is a version of the synthetic-data story that ends too early. You need labelled images, you have none, so you write a generator, you produce a corpus, you train a model, and you report a metric. Told that way, the generator is a means to a dataset and the dataset is the thing that mattered. For a one-off benchmark, that framing is fine. For an industrial computer-vision programme that has to serve a large, messy, heterogeneous real archive over years, it is the wrong unit of account, because it treats the generator as a step you finish rather than as the part of the stack whose reach governs everything downstream.

This whitepaper argues the other framing. The generator behind our raster well-log digitiser is infrastructure: a standing capability whose coverage of the archive's real variation is a reusable asset, and whose breadth, rather than the count of real labels, is what let a single fixed model generalise across the whole corpus of scans. The distinction is not semantic. It changes how you budget the work, how you staff it, how you decide when it is done, and what you defend when someone asks why you did not just annotate real data. The asset is the coverage. The model is a consumer of that coverage. Get the coverage wide enough and one model spans the archive; leave it narrow and no amount of downstream cleverness recovers the reach you never generated.

Two adjacent questions are easy to confuse with this one, and we set them aside deliberately. The first is the internal design of the generator, the geometry that draws a curve at native resolution, the annotation layer that stamps the printed furniture a scan carries, the specific knobs and their bounds. That is a genuine engineering subject, and it is not this one; it is the design of the machine, whereas this document is about what the machine is for. The second is the architecture of the model, the encoder-decoder with a

transformer refinement on the bottleneck that consumes the corpus. That too is its own subject. Here the model is held fixed on purpose, because the whole point of the infrastructure claim is that a fixed consumer's reach is set by the coverage it was fed.

The reason to insist on the separation is that the three questions have different owners, different time horizons, and different failure modes, and collapsing them hides the one that actually governs the programme. The model can be tuned in an afternoon and its gains are bounded; the architecture can be swapped and the corpus still serves the replacement; but the coverage of the generator is the slow-moving asset that outlives both, and a gap in it cannot be closed by retraining or by a better backbone. When a programme underperforms and the team reaches first for a bigger model or a longer schedule, it is usually reaching past the binding constraint, which is that the generator never covered the part of the archive the model is failing on. Naming coverage as its own question, owned by a team on its own horizon, is what keeps that constraint visible instead of buried inside a modelling ticket.

Why the archive forces the reframing

The archive is what makes the platform view the correct one rather than a stylistic preference. We had 136,771 TIF scans and 7,781 LAS files to serve, spanning decades of scanning practice, paper condition, print convention and instrument layout. Nobody had ever annotated which pixel belongs to which curve, and nobody was going to, because hand-labelling at that scale is the exact bottleneck the programme exists to remove. So real labels were not a small resource to be spent carefully. They were zero. When the label budget is zero, the only thing standing between the model and the archive is how much of the archive's variation you can manufacture, which is to say the generator's coverage. That is why coverage, not annotation, is the load-bearing quantity, and why the generator that produces coverage is the asset worth treating as infrastructure.

II

THE ASSET

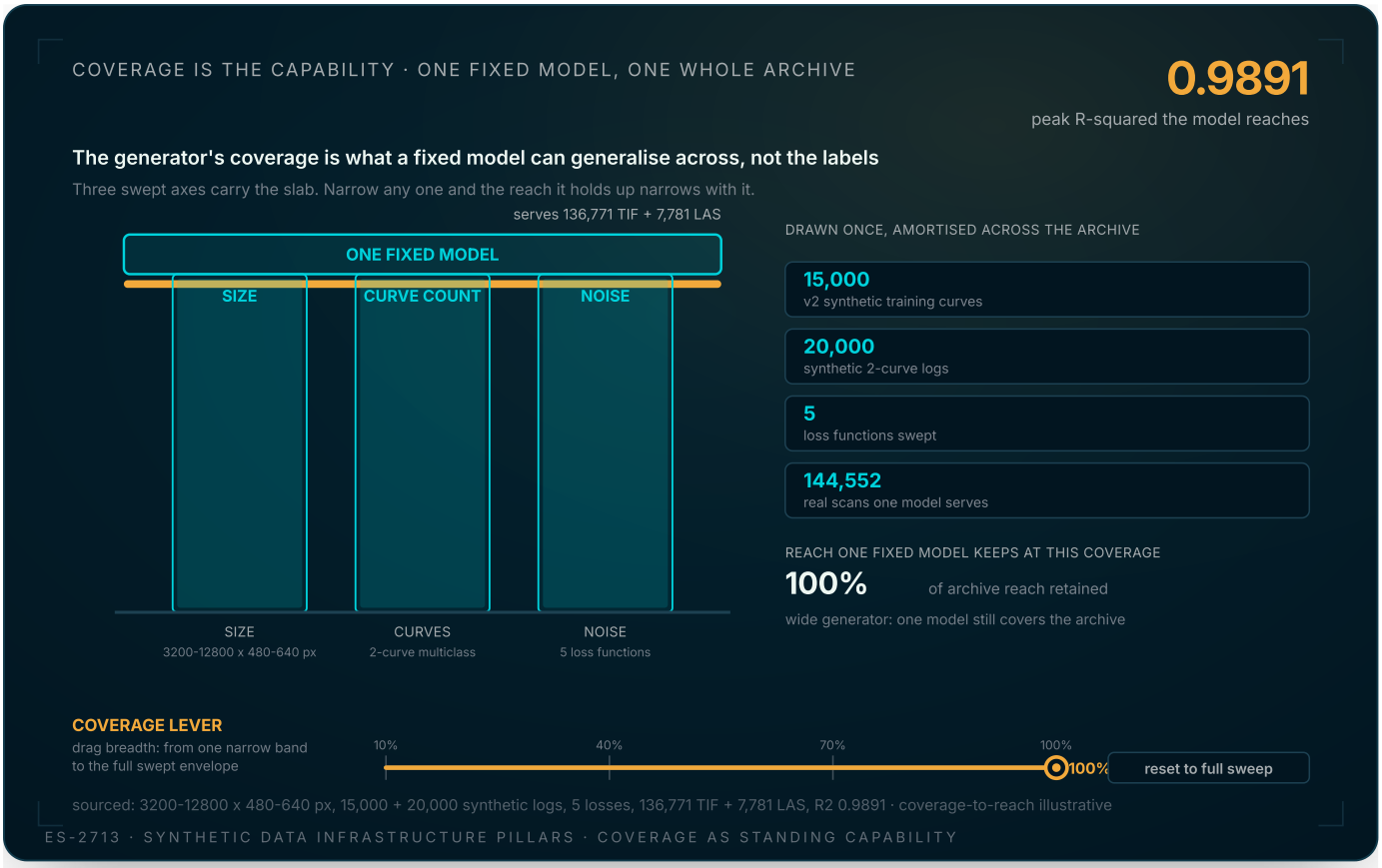
Coverage is the thing you are actually building

Three axes, each bounded by a real statistic

Coverage is not a single number; it is breadth along the axes on which real scans differ. For raster logs, three axes carry almost all of the variation that a segmentation model has to survive, and the discipline that makes the corpus honest is that each axis is bounded by a real archive statistic rather than by a convenient default.

The first axis is size. Real scanned logs are extreme in aspect ratio: our generator sweeps image width from 3,200 to 12,800 pixels against a height that runs only 480 to 640. That is not an arbitrary range. It is the range the archive actually spans, and a model that has only ever seen the narrow end of it will meet a 12,800-pixel scan at serve time as an out-of-distribution object. The second axis is curve count. The final multiclass corpus was built at two constant curves per log, the setting the deliverable required, and the number of curves changes the crossing and occlusion geometry the model must disentangle. The third axis is scan noise, the catch-all for the degradation and printed clutter a real scan carries, and the axis we probed hardest through the choice of training objective: the corpus was evaluated across 5 loss functions precisely because how a model tolerates foreground scarcity and thin-structure noise is a property you have to select for, not assume.

The reason to treat these three as the axes of an asset, rather than as generator settings, is that together they define a volume, and the fixed model can only reach the part of the real archive that falls inside that volume. This is the domain-randomisation result stated as an infrastructure principle: randomise the nuisance parameters widely enough and the real world reads as one more sample of the synthetic distribution [1], and the evidence across domains is that it is the breadth of that randomisation, not the photorealism of any single render, that makes the corpus transfer [2]. Coverage is therefore something you build on purpose, and its width is the asset's value.



Synthetic data read as a standing capability rather than a one-off dataset. Three pillars carry a single slab: one fixed model serving the whole real archive of 136,771 TIF and 7,781 LAS scans. Each pillar is one axis the generator sweeps and each is bounded by a real archive statistic: size (width 3,200 to 12,800 pixels, height 480 to 640 pixels), curve count (the 2-curve multiclass setting the corpus was built at), and noise (the 5 loss functions the corpus was evaluated against). The coverage lever drags generator breadth from one narrow band to the full swept envelope; the orange reach band under the slab and the peak-R-squared read-out fall together as coverage narrows, because a fixed model generalises across the archive only as far as the generator's coverage reaches. The 15,000 v2 synthetic training curves, the 20,000 synthetic 2-curve logs, the pixel dimensions, the 5 losses, the 136,771 TIF and 7,781 LAS counts, and the peak R-squared of 0.9891 are sourced from the engagement archive; the coverage-to-reach curve the lever traces is an illustrative monotone that treats coverage as an upper bound on reach.

The instrument makes the structural claim visible. Three pillars, one per axis, hold up a single slab: one fixed model serving the whole archive of 136,771 TIF and 7,781 LAS scans. Each pillar is bounded by its real statistic, the size band, the curve count, the noise sweep across 5 losses. The coverage lever narrows the generator from the full swept envelope toward a single band, and the orange reach bar under the slab shrinks with it, because a fixed model generalises across the archive only as far as the generator's coverage reaches. Pull coverage down far enough and the model no longer covers the archive at all. That is the whole argument for why coverage is the asset: the slab is only as wide as the pillars that carry it.

HOW WE TREAT COVERAGE IN PRACTICE

Read every coverage decision against the widest, oddest members of the real archive, not the median scan, because the extreme members are what set the axis bounds. A generator dialled to the average is a generator that fails at serve time on exactly the scans a human would also find hard. The bound on each axis is a real archive statistic or it is a guess, and a guess is a hole in the coverage the model will fall through later.

What coverage buys that volume alone does not

It is worth being precise about why coverage, rather than sheer image count, is the asset. You can generate a million near-identical logs and cover almost nothing; you can generate fifteen thousand well-spread ones and cover the archive. The empirical literature on data scaling is often read as a volume argument, but the mechanism underneath it is coverage: performance improves as the training distribution comes to span more of the space the model will be tested on [3]. A generator is the one tool that lets you buy coverage directly, by widening the randomisation rather than by collecting more of the same. That is the property that makes it infrastructure rather than a dataset: a dataset is a fixed sample of a distribution, while a generator is the distribution, tunable, and you can always widen it when a new corner of the archive shows up.

The 15,000-curve figure is worth dwelling on for what it is not. It is not the largest corpus we could have drawn; the generator that produced it had already emitted 20,000 two-curve logs, and it could have emitted ten times that at the cost of disk and wall-clock alone. The version-two corpus settled at 15,000 because that count, spread across the swept size, curve-count and noise ranges, was enough to span the archive, and adding near-duplicates past that point buys volume without buying coverage. This is the practical difference between the two accounting units. If you are counting images, more is always nominally better and you never know when to stop. If you are counting coverage, you stop when the axes are spanned to their real bounds, and you spend the next unit of effort widening a bound rather than deepening a sample. The whole reason the corpus could be that small and still carry the archive is that every render was placed to extend coverage rather than to pad a count, which is a property of how the generator was dialled, not of how long it ran.

There is a corollary that matters for anyone reasoning about cost. Because coverage rather than volume is the asset, the marginal synthetic image is worth almost nothing once an axis is already spanned, and worth a great deal when it opens a corner of an axis the corpus had not reached. That non-uniform value is invisible if you price synthetic data per image and obvious the moment you price it per unit of coverage. It is also why a generator, once built, keeps returning value long after the corpus is drawn: the next real operator whose scans sit in an uncovered corner is served not by collecting their labels but by widening one bound and redrawing, which is a capability the generator has and a static dataset never will.

III

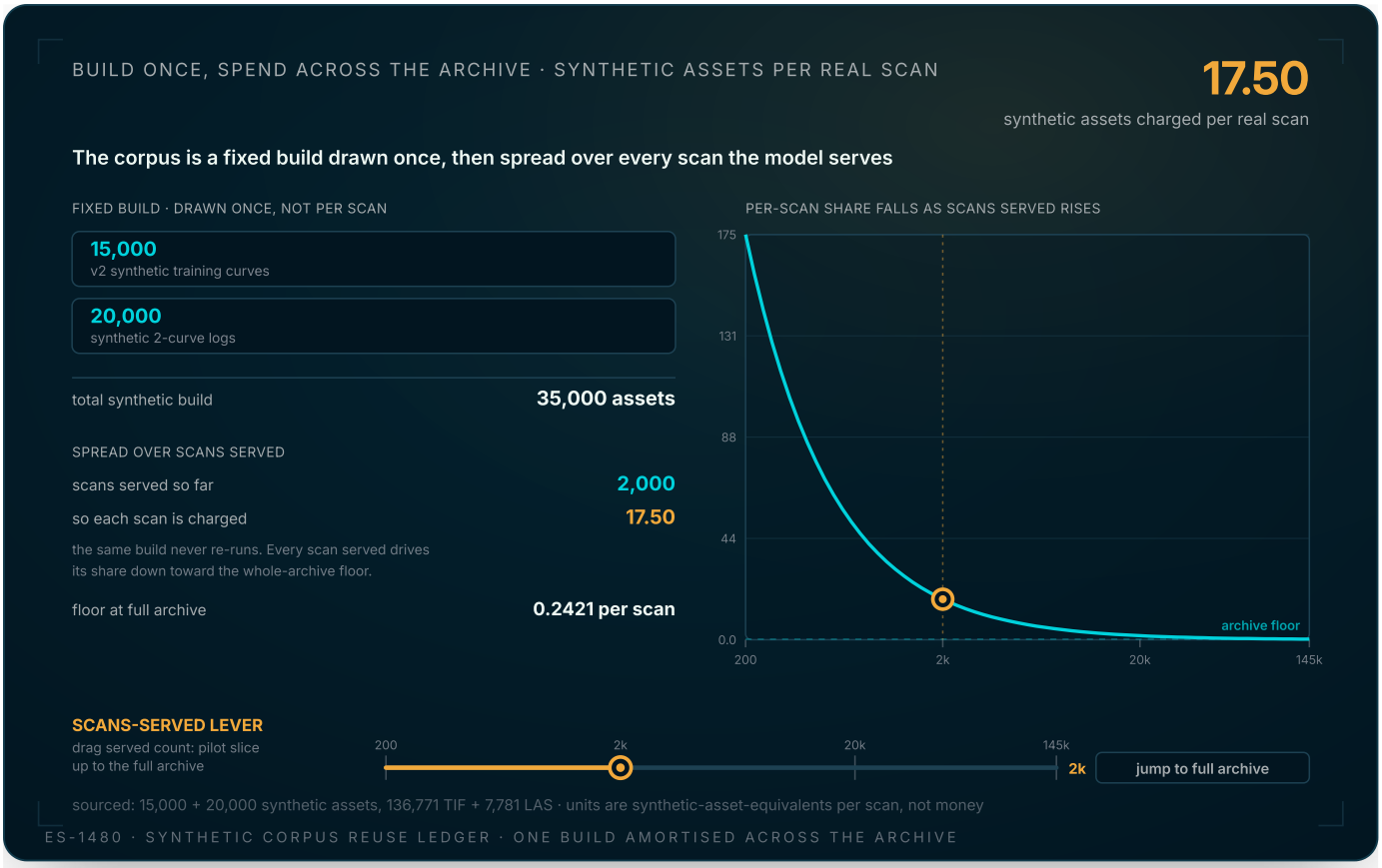
THE ECONOMICS

Build once, spend across the whole archive

A corpus is a fixed build that amortises

The second half of the infrastructure argument is economic, and it is the half that survives contact with a finance conversation. A synthetic corpus is a fixed build cost. You pay to design the generator and to draw the corpus once: 15,000 synthetic training curves in the version-two build, taken from a generator run that produced 20,000 two-curve logs. That build does not repeat when the model serves a new real scan. Inference against a real TIF is a forward pass, not a regeneration of the corpus. So the natural way to account for the corpus is not as a per-scan cost but as a fixed asset whose cost per real scan falls as the model serves more of the archive.

This is amortisation in the plainest sense, and it is the reason a synthetic-data capability behaves like infrastructure on the books. Spread a fixed build across a first pilot slice of a few hundred scans and its per-scan share is large. Spread the same build across all 144,552 real scans in the archive and the per-scan share collapses toward a floor. The corpus was expensive to stand up once and nearly free per scan thereafter, which is exactly the cost shape of a road or a pipeline, not the cost shape of a consumable [5].



The infrastructure claim stated as a cost that amortises. A synthetic corpus is a fixed build drawn once, the 15,000 v2 synthetic training curves plus the 20,000 synthetic 2-curve logs, and that build never re-runs. The scans-served lever drags the denominator from a first pilot slice up to the full real archive of 136,771 TIF and 7,781 LAS scans, and the orange marker slides down the ladder as the synthetic-build share charged to each real scan falls toward the whole-archive floor. That falling share is what makes procedural generation a standing asset rather than a per-job expense: the more of the archive one fixed model serves, the less of the one-time build each scan carries. The synthetic-asset counts and the real-archive counts are sourced from the engagement archive; the share is expressed in synthetic-asset-equivalents per real scan, not in money, and the served-count sweep is the derived quantity.

The ledger reads that economics directly. The fixed build, 15,000 plus 20,000 synthetic assets, sits in the numerator and never re-runs. The scans-served lever drags the denominator from a first pilot slice up to the full 144,552-scan archive, and the orange marker slides down the amortisation ladder as the synthetic-build share charged to each real scan falls toward the whole-archive floor. The shape is the argument: a build that looks expensive against a pilot slice is nearly free against the archive, which is the difference between a per-job expense and a standing asset. It is also why the right time to widen coverage is early, while the build cost is still being amortised against a growing denominator rather than being re-incurred per delivery.

The staffing consequence of the economics

If the corpus is infrastructure, the team that owns it is a platform team, not a delivery team, and that has a concrete staffing consequence. A delivery team is sized to the number of jobs; a platform team is sized to the breadth of the capability and the rate at which the archive throws new variation at it. The work that matters is widening coverage when a new operator's scans expose an uncovered corner, tightening the axis bounds against fresh archive statistics, and keeping the generator honest so it does not quietly teach the model its own habits. None of that is per-scan work. All of it compounds: every widening of the generator raises the ceiling for every model that will ever consume it, which is the return profile of infrastructure spend rather than of piecework.

> **“We stopped counting the corpus as a cost of each digitisation job and started counting it as a fixed asset the whole archive draws down. That single change in accounting is what made the case for widening coverage instead of collecting labels.”**
>
> — FROM OUR OWN DELIVERY NOTES

IV

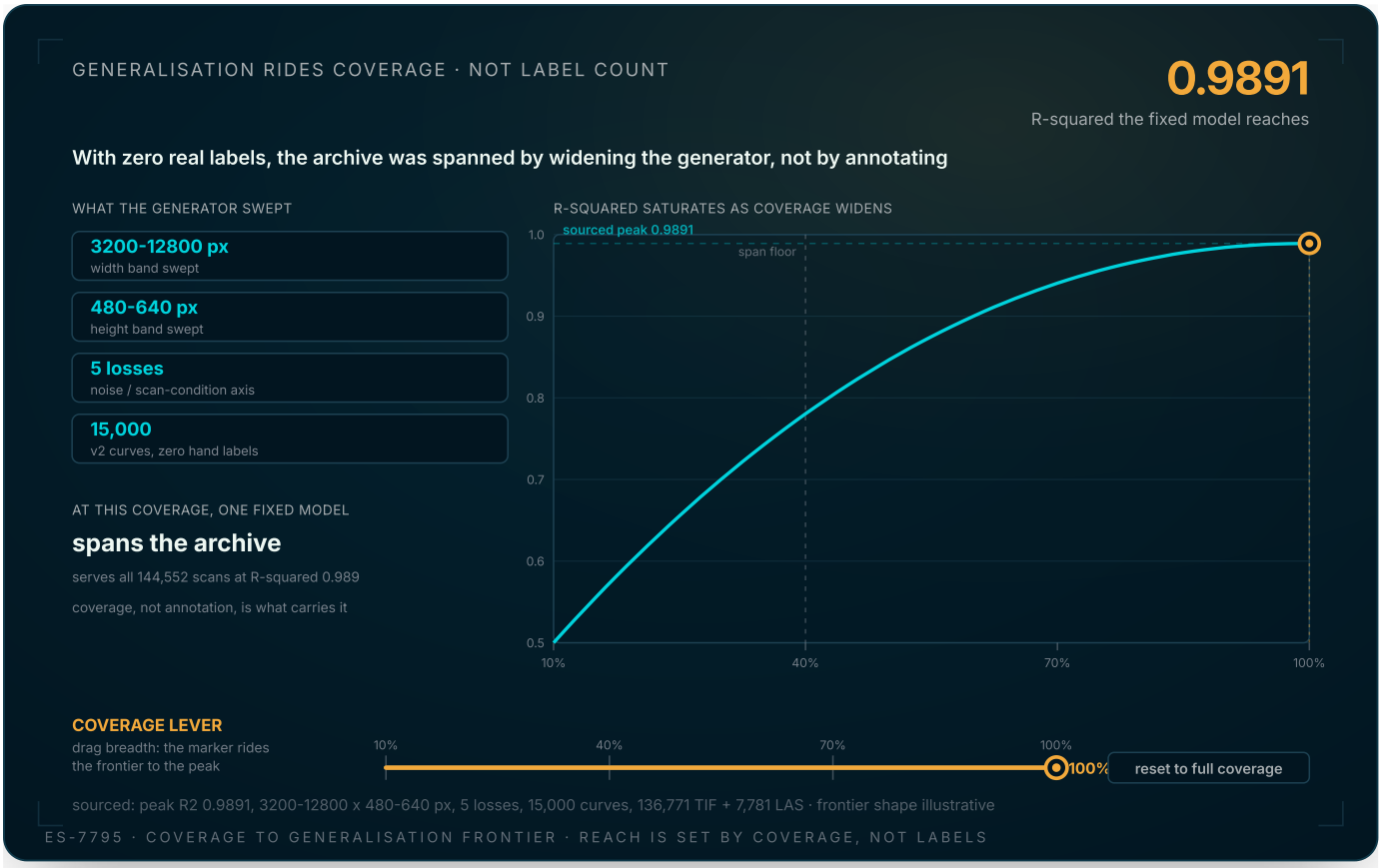
THE MECHANISM

Reach is set by coverage, not by label count

The claim with teeth: labels were not the lever

The infrastructure framing would be a comfortable story and nothing more if it did not make a falsifiable claim about where generalisation comes from. It does. The claim is that the reach of the fixed model across the real archive was set by the coverage of the generator and not by the volume of real labels, and the reason we can state that so plainly is that the volume of real labels was zero. The 15,000-curve corpus that carried the digitiser to a peak reconstruction R-squared of 0.9891 contained no human annotations at all. Whatever produced that generalisation, it was not annotation, because there was none to produce it.

That is the sharp end of the platform view. In the ordinary supervised story, you can always argue that a better number would come from more labelled data, and the generator is a stopgap until you can afford to annotate. Here that argument has nothing to stand on. The only knob that touched the archive was coverage, so the generalisation the model reached is a reading of how wide the generator was dialled. Widen the coverage and the reachable fraction of the archive grows; narrow it and the model falls off the parts of the archive the generator stopped covering, and no quantity of real labels would put those parts back, because the failure is one of distribution, not of sample size.



The mechanism under the infrastructure claim, stated as a frontier. A fixed model's generalisation across the real archive, read as R-squared, is set by how much of the archive's dimensional variation the synthetic generator covered, not by how many real labels were collected, because there were none: the 15,000 v2 training curves carry zero hand labels. The coverage lever drags generator breadth and the orange marker rides the frontier, saturating toward the sourced peak of 0.9891 as coverage widens and sliding toward the floor as it narrows. A vertical span-floor guide marks the coverage below which one fixed model can no longer span the whole archive of 136,771 TIF and 7,781 LAS scans; past that line, more real labels would not help, only wider generator coverage. The peak R-squared, the width and height bands, the 5 loss functions, the synthetic-curve count, and the archive counts are sourced from the engagement archive; the frontier shape that maps coverage to R-squared is an illustrative saturating curve anchored on the sourced peak.

The frontier states the mechanism as a curve. Reconstruction R-squared is plotted against the fraction of the archive's variation the generator covers, and the orange marker rides that frontier up toward the sourced peak of 0.9891 as coverage widens. The vertical span-floor guide marks the coverage below which one fixed model can no longer span the whole archive; to the left of it, the model simply cannot serve the corners the generator stopped reproducing, and more real labels do not move the line because the gap is a coverage gap. This is the picture that turns the infrastructure framing from a metaphor into an operating rule: when the number is short, the question to ask is which axis of the generator is too narrow, not how many real labels are missing.

Why this does not collapse into overfitting the generator

A fair objection is that coverage-driven generalisation risks teaching the model the generator's own regularities rather than the geoscience, and that a high synthetic number could be an artefact of a model that has learned to read synthetic logs specifically. That risk is real, and it is exactly why coverage has to be breadth rather than volume, and why the deliverable is graded on reconstructed-curve agreement against real ground truth rather than on the synthetic validation split alone. A generator dialled narrow does invite the model to memorise its habits; a generator dialled wide, across the size, curve-count and noise axes bounded by real statistics, removes the habits to memorise, because there is no single synthetic regularity left to latch onto once the nuisance parameters are randomised across the archive's real range [1][2]. The defence against overfitting the generator is the same lever as the source of the generalisation: more coverage. That symmetry is the tell that coverage, and not annotation, is the mechanism in play.

V

THE PRACTICE

Standing up a synthetic-data capability as infrastructure

Budgeting, staffing and defending the capability

Reading the three arguments together gives a practical posture for any team standing up industrial computer vision under a no-labels constraint. Budget the generator as a capital item, not a per-job cost: it is built once and amortised across the archive, so the case for widening it is strongest early, while the denominator it is charged against is still growing. Staff it as a platform team whose work is coverage and honesty rather than throughput, because every widening of the generator raises the ceiling for every future model rather than clearing a single job. Define done by coverage against the real archive's axis statistics, not by a synthetic validation number, because the number that matters is the reconstructed-curve agreement on real scans and that is a function of how much of the archive the generator spans. And when someone asks why you did not simply annotate real data, the answer is not that annotation is expensive; it is that annotation is a fixed sample of the distribution while the generator is the distribution, and only the second can be widened to reach a corner of the archive you had not seen when you started.

Coverage is the asset

- The generator sweeps size, curve count and scan noise across the archive's real range
- Width 3,200 to 12,800 pixels and height 480 to 640, each bounded by a real archive statistic
- That swept envelope, not the label count, is what a fixed model can generalise across
- Narrow any axis and the reach the model keeps narrows with it

Reuse is the economics

- The 15,000-curve version-two corpus is a fixed build drawn once, from a 20,000-log run
- That build never re-runs per real scan; it amortises across the whole archive
- Per-scan share of the build falls toward a floor as more of the archive is served
- This is what makes procedural generation a standing asset, not a per-job expense

Coverage drives generalisation

- The corpus carried zero human annotations, so labels cannot be the lever
- Reconstruction reached a peak R-squared of 0.9891 on the served archive
- Below a coverage floor, a fixed model can no longer span the archive
- Past that floor, only wider generator coverage helps, not more real labels

What is reusable beyond well logs

Nothing in the argument is specific to well logs. The three-axis structure, size, instance count, and domain noise, is the general shape of coverage for any industrial imagery where the real corpus is heterogeneous and unlabelled: the axes rename to the ones that carry your archive's variation, but the claim holds that their swept volume sets the reach of any fixed model trained inside it. The economics rename identically: a generator is a fixed build that amortises across whatever population it serves, and the per-unit cost falls toward a floor set by the size of that population. And the mechanism is the same wherever real labels are scarce enough that coverage, not annotation, is the only knob that touches the target distribution. The reusable deliverable of this whitepaper is not a generator; it is the

discipline of treating one as infrastructure, budgeting its coverage as an asset, and defending its reach as the thing that carried a fixed model across an archive no one ever labelled.

WHAT TO CARRY OUT OF THIS

1. A procedural generator is infrastructure, not a delivery. Its coverage of **size**, **curve count** and **scan noise** is the reusable asset, and that swept volume sets the upper bound on how much of the archive one fixed model can serve.
2. Bound every coverage axis by a real archive statistic: width **3,200** to **12,800** pixels, height **480** to **640**, evaluated across **5** loss functions. A bound that is a guess is a hole the model falls through at serve time.
3. Account for the corpus as a fixed build, not a per-scan cost. The **15,000**-curve version-two corpus, drawn from a **20,000**-log run, is drawn once and amortises across all **144,552** real scans toward a floor.
4. Reach was set by coverage, not labels. The corpus carried **zero** human annotations and the digitiser still reached a peak reconstruction R-squared of **0.9891**, so annotation cannot be the mechanism, coverage is.
5. When the number is short, widen the generator, do not collect labels. Below a coverage floor a fixed model cannot span the archive, and past that floor only wider coverage helps, which is the operating rule the infrastructure view gives you.

Limitations

The infrastructure framing rests on a specific programme and its metric, and the claims should be read inside those bounds. The peak reconstruction R-squared of 0.9891, the 15,000-curve version-two corpus, the 20,000-log generator run, the 3,200 to 12,800 by 480 to 640 pixel dimensional bounds, the 5 loss functions, and the 136,771 TIF and 7,781 LAS archive counts are sourced from the engagement archive; the peak is a best-case example on the multiclass task, not an average across every curve and scan, and a single headline R-squared should not be read as uniform quality over the whole archive. The three instruments argue the structure of the claim, and the specific response curves in them are illustrative rather than measured: the coverage-to-reach relationship in the pillars, the exact

position on the amortisation ladder between the pilot slice and the archive floor, and the shape of the coverage-to-generalisation frontier are all schematics anchored on the sourced endpoints, built to argue the proportional effect rather than to predict a point value. The claim that coverage rather than label count is the mechanism is grounded in the fact that the corpus had zero human labels, which makes annotation unavailable as an explanation for this programme; it is not a general proof that labels never help, only that on this archive they were not the lever available. The three coverage axes are the ones that carried our variation, and a different industrial-imagery corpus may be dominated by axes we did not have to model. Finally, coverage-driven generalisation is only safe when the deliverable is graded against real ground truth, as ours was on reconstructed-curve agreement; a synthetic validation number read in isolation can flatter a model that has learned the generator rather than the target, and the guard against that is breadth of coverage plus a real-data grade, not the synthetic score alone.

References

1 Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P. (2017). *Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World.* IROS. The principle that a generator randomised widely enough over its nuisance parameters makes the real world read as one more sample of the synthetic distribution. https://arxiv.org/abs/1703.06907

2 Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S., Birchfield, S. (2018). *Training Deep Networks with Synthetic Data: Bridging the Reality Gap by Domain Randomization.* CVPR Workshops. Evidence that breadth of randomisation, not photorealism, is what makes a synthetic corpus transfer. https://arxiv.org/abs/1804.06516

3 Sun, C., Shrivastava, A., Singh, S., Gupta, A. (2017). *Revisiting Unreasonable Effectiveness of Data in Deep Learning Era.* ICCV. The empirical case that model performance scales with the coverage and volume of training data, which is the lever a generator controls directly. https://arxiv.org/abs/1707.02968

4 Ronneberger, O., Fischer, P., Brox, T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation.* MICCAI. The encoder-decoder segmentation shape the fixed model is built on, trained once and served against the whole archive. https://arxiv.org/abs/1505.04597

5 Nikolenko, S. I. (2019). *Synthetic Data for Deep Learning.* arXiv. A survey framing synthetic data as a reusable resource for training when real labels are absent or expensive, which is the infrastructure view this whitepaper adopts. https://arxiv.org/abs/1909.11512

6 Koroteev, D., Tekic, Z. (2021). *Artificial intelligence in oil and gas upstream: Trends, challenges, and scenarios for the future.* Energy and AI. Context for why a large legacy raster archive is worth standing up a durable digitisation capability for in the first place. https://www.sciencedirect.com/science/article/pii/S2666546820300033

Get the full whitepaper

This page is the long-form summary. The complete whitepaper adds the per-axis coverage map against the real archive statistics, the amortisation schedule from the pilot slice to the full archive, the coverage-versus-generalisation evidence on the held-out real curves that grounds the 0.9891 peak, and the operating checklist for budgeting and staffing a synthetic-data capability as durable infrastructure.

EARTHSCAN

Synthetic Data as Infrastructure for Industrial Computer Vision

Published February 2023. Generated 2026-07-22.
https://www.earthscan.io/whitepapers/synthetic-data-as-infrastructure-for-industrial-computer-vision