Ownership Models for the AI You Just Built

When a multi-year subsurface-AI R&D engagement ends, the operator is holding a working model and a decision most programmes defer until it is urgent: who owns the compute that will train the next version, and the one after that. The model is portable; the machine that makes new models is not, and the choice of who runs that machine sets the operator's cost base, its staffing, and how fast it can respond to new wells for years. This whitepaper generalises the transition menu we put in front of a mid-sized Middle East carbonate operator at that gate into three ownership models plotted on one curve. Tier one, replicate the stack in-house, keeps full control but requires the operator's own ICT to grow AI muscle and runs a full-model retrain in two-to-three weeks on hardware the operator buys and staffs. Tier two, a light or medium managed service on the operator's premises, lowers cost and internal lift and stretches a full retrain to days. Tier three, as-is managed supercompute, is zero lift-and-shift because the R&D engagement already paid the upfront investment, so a full retrain lands in minutes-to-hours for a fixed annual fee. We argue three things with three instruments. First, that per-model training time and internal-capability burden move in opposite directions across the tiers, which is why the decision is a curve and not a ranking. Second, that the train-time collapse is bought by climbing three orders of magnitude of compute, from a workstation GPU stack through a DGX A100 node to a multi-node SuperPod, and that only the fixed-fee tier puts the top rung under a retrain without a capital purchase. Third, that which tier fits is decided by three concrete questions about the operator's ICT rather than by the fastest train-time on offer. The deliverable is a way to price the R&D-to-industrialization gate as the ownership decision it is, before the last progress report is signed.

Tarry Singh
Founder & CEO

Narendra Patwardhan
Research Collaborator

COMPUTE-OWNERSHIP / MANAGED-SERVICE / SUPERCOMPUTE / DGX-A100 / TRAIN-TIME / INDUSTRIALIZATION

CONTENTS

The model was the easy part to hand over. It fits in a container. It has a checkpoint file, a config, and a page of numbers that say how well it picks fractures and beddings on a borehole-image log. An operator can copy it to a laptop and, with a little help, run it against a well it has never seen. What does not fit on a laptop, and what almost no R&D programme scopes explicitly, is the machine that makes new models. When the next batch of wells arrives, when the geology shifts to a formation the training set under-covered, when a reviewer asks for a retrain with a different loss weight, someone has to own the compute that turns that request into a new checkpoint. Deciding who owns that compute is the real transition decision, and it is usually made late, under time pressure, as the last progress report is being signed.

We ran that decision live with a mid-sized Middle East carbonate operator at the end of a multi-year engagement. The technical work was a set-prediction model for fracture and bedding picking on borehole-image logs, but the ownership question it raised is general: any operator that has just paid for an AI R&D programme faces the same menu at the R&D-to-industrialization gate. This whitepaper is about that menu. It sets out three ownership models and argues that they are not three points on a ranking where higher is better, but three points on a curve where two things move in opposite directions. As you move from owning everything to owning nothing, the time to train a full model collapses from weeks to minutes, and the cost and internal-capability burden trade the other way. The audience is the chief technology officer, the CIO or head of ICT, and the project director who together sign for what the operator will run after the researchers leave.

The thing you are actually deciding

It helps to be precise about what is on the table, because the ownership decision is easy to confuse with two decisions it is not.

It is not the decision to build capability in-house versus rent it per project. That is a prior, strategic question, and we have written about the phased operating model that answers it in "A Phased Operating Model for Standing Up Subsurface AI Capability" and about the capability handover itself in "Handing Over the Keys." Assume that question is settled: the operator wants to keep training models after the engagement ends. The ownership decision sits one level down. Given that you will retrain, on whose compute, run by whom, does the retrain happen.

It is also not the decision about the model's architecture or its accuracy. Those are fixed by the time you reach this gate. The checkpoint is what it is; the ownership tiers all run the identical model. What changes across the tiers is the wall-clock time and the organisational cost of producing the next checkpoint from new data.

> **THE DECISION IN ONE LINE**
>
> You are choosing who owns the machine that makes new models, not the model itself. The model is portable and settled; the retrain loop is neither, and the tier you pick sets its speed and its cost for years.

Why does this matter enough to warrant a whitepaper. Because the retrain loop is where the running cost of a deployed model lives. A survey of machine-learning deployment case studies makes the general point plainly: the effort and cost of an ML system are dominated by what happens after the first model ships, not by the initial build. A subsurface model is no exception. New wells arrive on their own schedule, label conventions drift, and each material change is a retrain. The tier that owns your retrain loop therefore owns a recurring line in your budget and a recurring dependency in your operations. Getting it wrong is not a one-time mistake; it is a mistake you pay for every time the geology gives you a reason to retrain.

What actually triggers a retrain

The abstract case for a fast retrain loop is easy to nod along to and easy to under-weight. It becomes concrete only when you look at what forced retrains during the R&D phase itself, because those same forces do not stop when the researchers leave. Three of them are worth naming, because each one is a retrain the operator will have to run on whatever compute it ends up owning.

The first is simply more wells. The programme's own model iteration shows the effect directly: stepping the training set from a handful of wells to the full set moved fracture recall at a five-centimetre depth tolerance from roughly forty percent to seventy-five percent. More labelled wells was the single largest lever on quality across the whole engagement, larger than any architectural change. That is not a one-time gain to be banked and

forgotten. Every time the operator drills and interprets new wells in a formation the current model under-covers, the same lever is available, and pulling it means a retrain. An operator that cannot retrain cheaply is an operator that leaves that quality on the table.

The second is label style. The programme learned, expensively, that more data is not automatically better data. Adding a fifteenth well whose picks were made in a different interpretive style actually dropped validation performance relative to the fourteen-well model, because the label convention differed from the rest of the set. The lesson is that the retrain loop is not only fed by new wells; it is fed by decisions about which wells and which labels to include, and those decisions are made by the operator's own interpreters after handover. Each such decision is a retrain-and-compare, and comparing two candidate training sets honestly means running the training twice.

The third is the synthetic-data budget. During the engagement the team grew roughly nine hundred image-and-ground-truth pairs to well over fifty thousand through overlap and augmentation, and then discovered that too much synthetic data degraded the model rather than improving it; the fix was to halve the augmentation and increase the patch stride as real wells came in. Tuning that balance is an ongoing activity, not a settled hyperparameter, and every adjustment to it is another retrain. The point of these three examples is not the specific numbers. It is that the retrain trigger is not exotic. It is the normal texture of running a subsurface model against a live, growing archive, and it is why the speed and cost of the retrain loop is the thing the ownership decision is really setting.

Three tiers, one curve

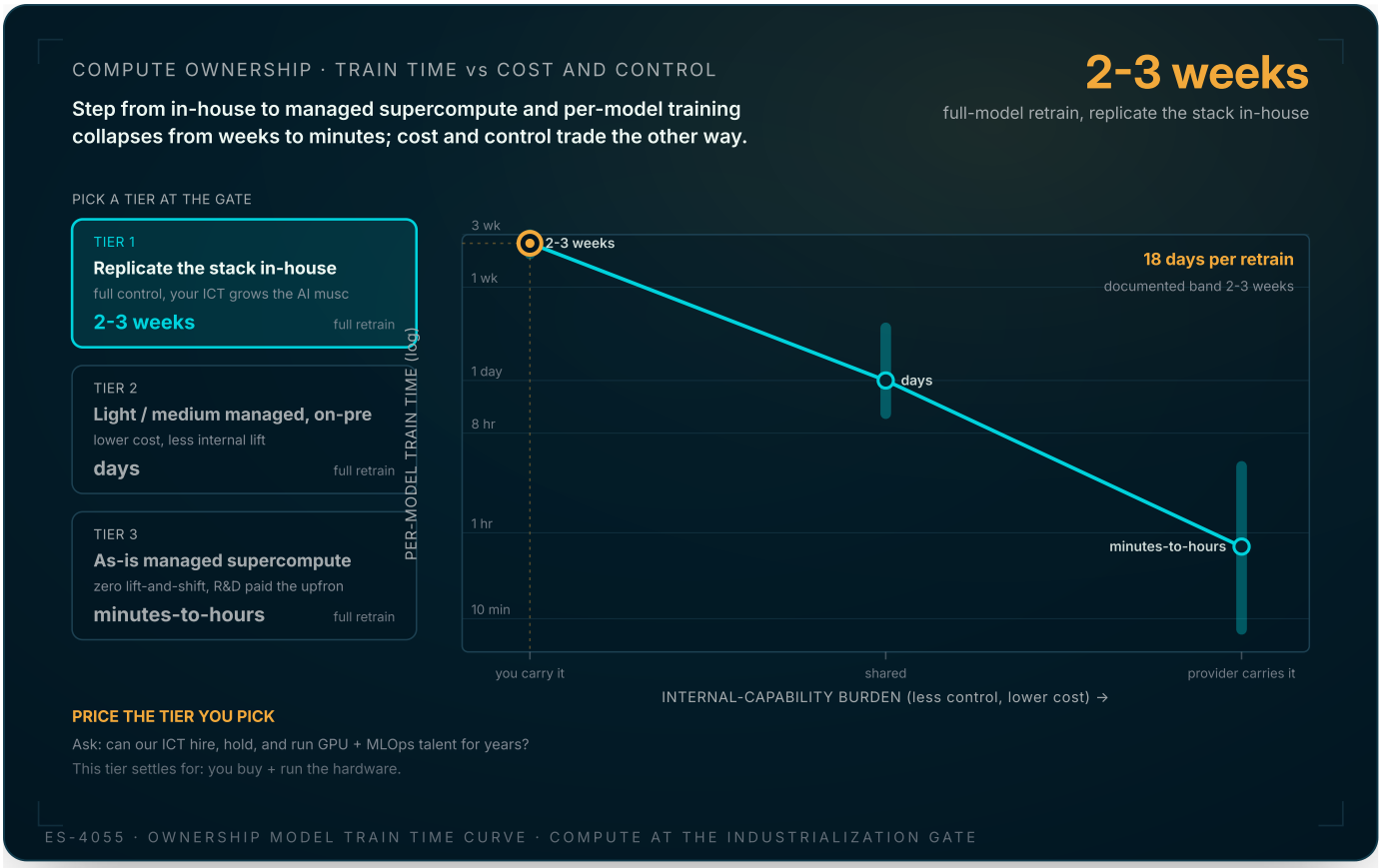
The menu, stated as the operator sees it at the gate, has three entries.

Tier one: replicate the stack in-house. The operator stands up the full training environment on hardware it owns, and its own ICT staff learn to run it. This is the maximum-control option. Nothing leaves the perimeter, the operator answers to no external provider, and the capability is genuinely internal. The price of that control is capability burden. The operator's ICT has to grow real AI muscle: GPU operations, an MLOps toolchain, the ability to reproduce a training run and debug it when it diverges. And on hardware sized to what a mid-size operator will realistically buy, a full-model retrain runs in two-to-three weeks. That is not the model being slow; it is a training job that wants a lot of parallel compute running on a modest amount of it.

Tier two: a light or medium managed service on the operator's premises. A provider stands up and runs the training environment inside the operator's data centre. The hardware and the operations are a managed service; the data never leaves the operator's premises. This lowers both cost and internal lift, because the operator is not hiring and holding the full GPU-and-MLOps team itself, and it puts more capable hardware under the retrain than the in-house baseline typically buys. A full-model retrain here stretches to days rather than weeks.

Tier three: as-is managed supercompute. The operator retrains on the provider's existing supercompute, as-is, with no lift-and-shift. This is the option the R&D engagement quietly pre-paid for: the upfront investment in the training stack, the cluster, the MLOps plumbing, was already made during the research phase, so the operator inherits it rather than rebuilding it. A full-model retrain lands in minutes-to-hours, and the commercial shape is a fixed annual fee rather than a capital line plus a staffing line. The illustrative figure from the year-two services frame we worked with was on the order of 313,000 USD per year, quoted here as a band, not a price.

Read those three as a list and the temptation is to rank them, with the fast one on top. That reading is wrong, and the chart below is built to show why. Plot per-model training time on a logarithmic axis against the internal-capability burden the operator takes on, and the three tiers fall on a single curve that descends as you move right. The fast tier is fast precisely because you carry the least; the controlled tier is slow precisely because you carry the most. There is no dominating option, only a trade you have to price.



The compute-ownership decision at the R&D-to-industrialization gate, drawn as one curve: per-model training time on a log axis against the internal-capability burden the operator takes on. Three sourced tiers sit on it. Replicating the stack in-house keeps full control but a full-model retrain runs 2-3 weeks on the operator's own hardware and its ICT has to grow AI muscle. A light or medium managed service on the operator's premises lowers cost and internal lift, and a full-model retrain stretches to days. As-is managed supercompute is zero lift-and-shift because the R&D engagement already paid the upfront investment, so a full retrain lands in minutes-to-hours for a fixed annual fee. Pick a tier on the left and the orange marker snaps to it: as you step right, less control and lower internal cost, per-model training collapses from weeks toward minutes. The weeks-to-days-to-minutes train-time ladder and the zero-lift-and-shift framing are sourced from the engagement's progress-report bundle; the exact minutes used to position each band and the 313,000 USD/yr fixed-fee figure are illustrative, read from the documented ladder and the year-2 services frame, and the internal-burden axis is an ordinal reading rather than a measured quantity.

The curve is the argument. Stepping from in-house replication to as-is supercompute buys roughly three orders of magnitude of train-time, from weeks to minutes, and the thing you spend to buy it is control and internal capability, not money in a simple sense. An operator that reads this curve and reaches straight for the minutes has not made a mistake yet, but it has to know it is trading away the muscle its ICT would have grown on the slow tiers, and that the muscle is worth something the day the provider is unavailable or the contract is up for renewal.

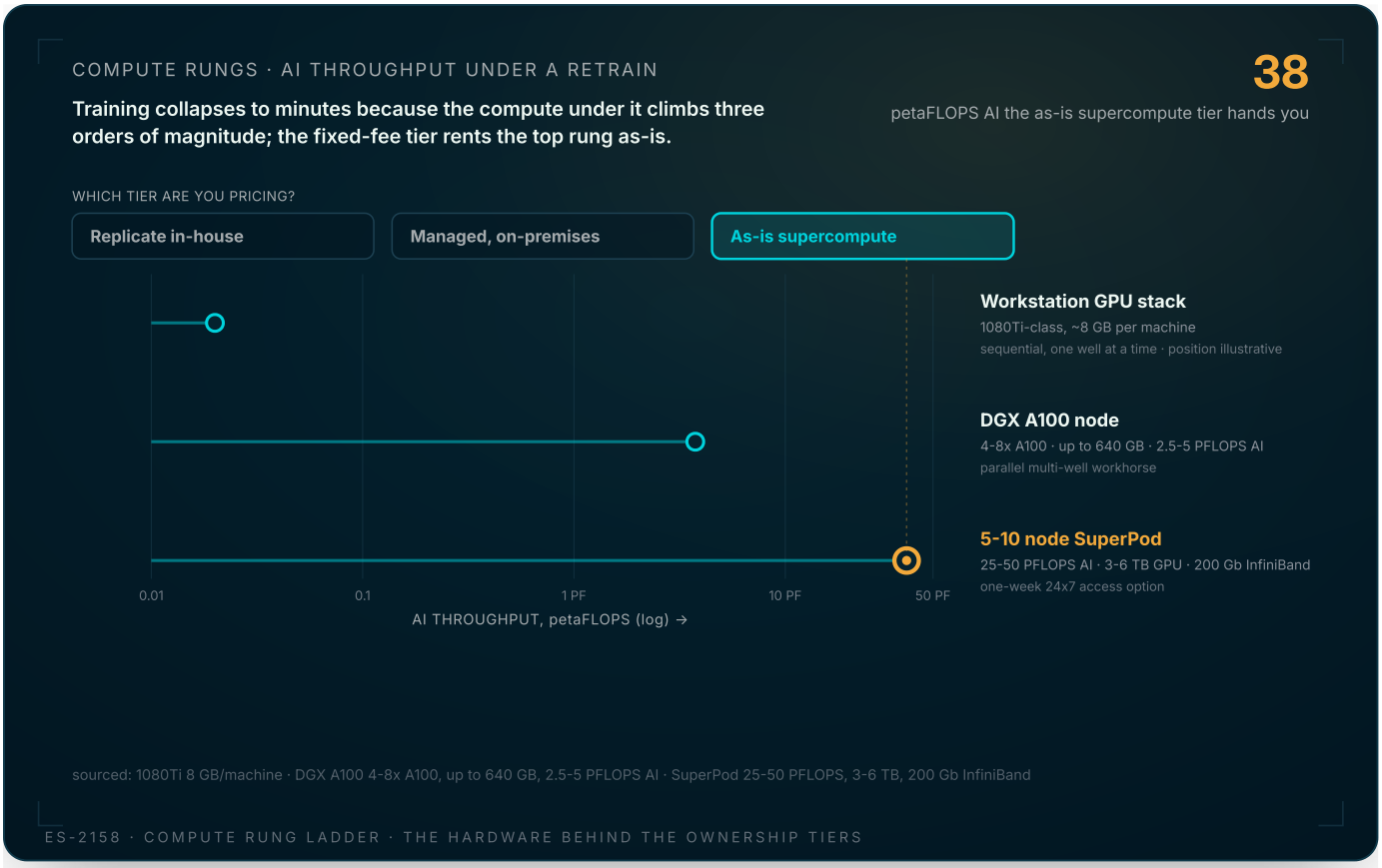
Why the fast end is fast

The train-time collapse is not magic and it is not the model getting cleverer between tiers. It is compute. To move a full-model retrain from weeks to minutes you have to put a great deal more parallel throughput under the job, and the three tiers correspond, roughly, to three rungs of a compute ladder that spans three orders of magnitude.

At the bottom is the kind of hardware an operator's ICT starts from when it replicates in-house: a workstation-class GPU stack, on the order of eight gigabytes of GPU memory per machine, running the pipeline one well at a time. It works, and it is fully owned, but it is sequential and it is small, which is why the in-house retrain is measured in weeks.

The middle rung is a DGX A100 node: four to eight A100 GPUs, up to 640 gigabytes of total GPU memory, and 2.5 to 5 petaFLOPS of AI throughput. This is the parallel multi-well workhorse, and it is what a managed on-premises service typically puts under the retrain. The step up in throughput is what turns weeks into days.

The top rung is a multi-node SuperPod: a five-to-ten-node DGX A100 cluster delivering 25 to 50 petaFLOPS of AI, three to six terabytes of GPU memory, and 200-gigabit HDR InfiniBand between nodes, with a negotiated option for a week of round-the-clock access. This is what turns days into minutes. The decisive point for the ownership decision is not that the SuperPod exists; it is that buying and standing one up is a capital and staffing commitment a single operator rarely justifies for its own retrain cadence. The as-is tier lets the operator stand on that top rung without buying it, because the provider already owns it.



The hardware behind the three ownership tiers, stacked as rungs on one AI-throughput axis in petaFLOPS on a log scale. A workstation GPU stack (1080Ti-class, about 8 GB per machine) is the sequential, one-well-at-a-time baseline an operator's own ICT starts from; its position on the throughput axis is illustrative because the archive quotes it as memory rather than petaFLOPS. A DGX A100 node (4-8x A100, up to 640 GB total GPU memory, 2.5-5 petaFLOPS AI) is the parallel multi-well workhorse. A 5-10 node SuperPod (25-50 petaFLOPS AI, 3-6 TB GPU memory, 200 Gb HDR InfiniBand, with a one-week 24x7 access option) is the top rung. Toggle which tier you are pricing and the orange pointer marks the highest rung that tier puts under a retrain without a capital purchase: in-house starts on the workstation rung, managed on-premises rents a DGX node, and only the as-is fixed-fee tier stands the operator on the SuperPod rung without buying it. The petaFLOPS values are sourced from the engagement's infrastructure records; the workstation rung's log-axis position is a nominal placement.

The ladder makes the ownership logic concrete. The workstation rung is fully owned and fully yours, and it is slow. The DGX rung is faster and, in the managed tier, sits on your premises without you having to buy and run it. The SuperPod rung is the only one that delivers minutes, and it is the one that makes least sense to own outright at a single operator's scale. That asymmetry is the entire case for the fixed-fee, as-is tier: it is the rung you most want under a retrain and least want on your balance sheet.

One more property of the ladder matters for the ownership decision, and it is easy to miss when you read it as a hardware list. The rungs are not just faster; they change what kind of training job is even practical. On the workstation rung the pipeline runs one well at a time, because that is what the memory allows, so a full-set retrain is a long sequential slog. The

DGX rung, with up to 640 gigabytes of GPU memory across its cards, lets the operator hold and train against many wells in parallel, which is not only faster but a different workflow: comparing two candidate training sets, or sweeping a hyperparameter, becomes something you do in an afternoon instead of something you schedule for a fortnight. The SuperPod rung, with its 200-gigabit interconnect between nodes, makes even a full sweep cheap in wall-clock terms. So the train-time collapse is not merely the same job going faster; it is the operator gaining the ability to ask more questions of its own data per unit of calendar. That is the quiet second dividend of moving right along the curve, and it is why an operator with an active, growing archive values the fast tiers more than the headline train-time number alone would suggest.

The regressed-parameter aside, kept short

One technical detail is worth stating because it explains why the retrain is a heavy compute job in the first place, and it is the only place this paper touches the model's internals. The picking model does not fit each fracture sinusoid with a classical curve fit at inference. It regresses the sinusoid's parameters directly. A single fracture on the unrolled borehole image, before normalisation, is a sine wave

FRACTURE SINUSOID ON THE UNROLLED BOREHOLE IMAGE

Formula LaTeX

Copy LaTeX

$$y(x) = A\,\sin(\omega x + \varphi) + \text{offset}$$

where the amplitude maps to dip, the phase to azimuth, and the vertical offset to depth. The model learns to emit those parameters end-to-end for every fracture and bedding plane in a patch at once, across a training set of full-well images that run to well over a billion pixels each. That is what makes a full retrain a job that wants petaFLOPS, and it is why the ownership tiers separate so cleanly by train-time. The mechanics of the set-prediction model and its bipartite-matching loss are the subject of "GeoBFDT: End-to-End Detection Transformers for Fracture and Bedding Picking in Carbonate Image Logs" and are not re-derived here; for this paper the equation is only present to show why the compute bill is real.

Pricing each tier with a question, not a number

The curve tells you what each tier buys. It does not tell you which tier fits, and the most common failure at this gate is to let the train-time number decide. Minutes look obviously better than weeks on a slide. But the tier that produces minutes also produces a standing dependency on an external provider and grows no internal muscle, and for some operators that is exactly wrong. The fit is decided by three questions about the operator's own ICT, and each question, answered honestly, points at a different tier.

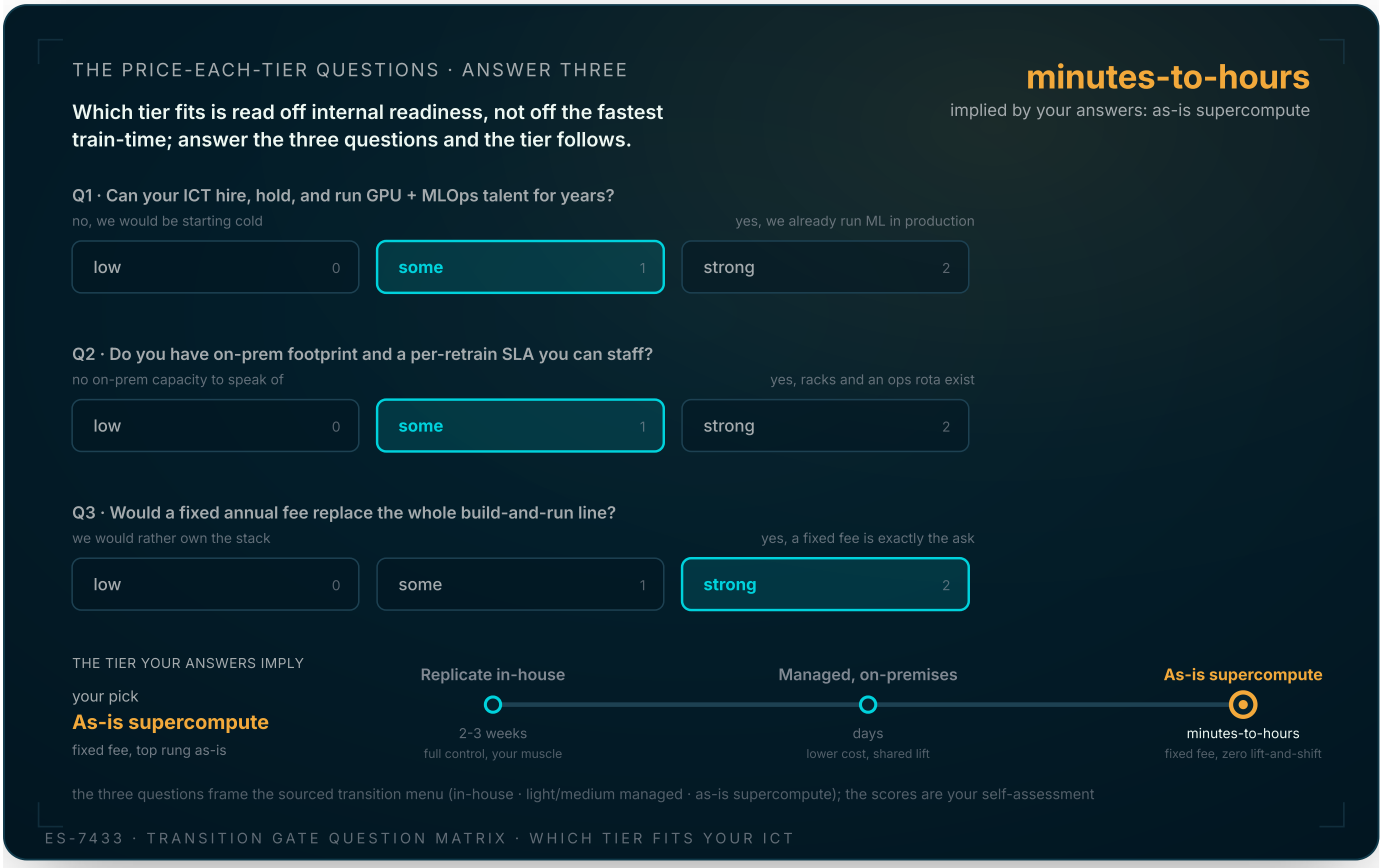
The first question is about talent. Can the operator's ICT hire, hold, and run GPU-operations and MLOps staff for years, not months. If the honest answer is yes, replicating in-house is viable and the two-to-three-week retrain is a price worth paying for full control and a genuinely internal capability. If the answer is no, in-house is a trap: the operator will buy the hardware, fail to staff it durably, and end up with a slow retrain loop that also does not work reliably.

The second question is about footprint. Does the operator have on-premises data-centre capacity and an operations rota that can carry a per-retrain service-level agreement. If yes, the light or medium managed on-premises tier is the natural middle: the data stays put, the hardware is more capable than the in-house baseline, and the operator staffs the interface rather than the whole stack.

The third question is commercial. Would a fixed annual fee that replaces the entire build-and-run line, the capital, the hardware, the team, be the right shape for this operator's budget. If yes, the as-is supercompute tier is not a compromise; it is the cleanest match, and it happens to be the fastest as well.

It is worth being blunt about why the commercial shapes differ so much across the tiers, because the fixed fee can look expensive next to a hardware quote until you count what the hardware quote leaves out. The in-house tier's true cost is not the GPU box. It is the box plus power, cooling, depreciation, and a standing team that can keep the training stack alive, and that team is the line item operators most consistently under-budget. Compute itself, on any of the tiers, is billed by the GPU-hour, and a subsurface programme's compute demand is lumpy rather than steady: it sits near idle between retrains and then spikes hard for the days or hours a full retrain runs. Owning enough hardware to make the spike fast means owning hardware that is idle most of the year, which is the worst possible utilisation profile to capitalise. The managed and as-is tiers exist precisely because a provider can

amortise that same hardware across many operators' spikes, so the operator pays for the spike rather than for a year of idle silicon. The fixed annual fee is the provider selling access to a utilisation curve the operator could never achieve alone.



The other half of the ownership decision. The train-time curve shows what each tier buys; this matrix shows what each tier costs you to run, so the pick is an honest match of internal capacity to tier rather than a reflex toward the fastest number. Answer three questions on a three-point scale: whether your ICT can hire and hold GPU and MLOps talent for years (which favours replicating in-house), whether you have on-premises footprint and a per-retrain SLA you can staff (which favours a managed on-prem service), and whether a fixed annual fee would replace the whole build-and-run line (which favours as-is managed supercompute). The orange marker lands on the tier your answers imply, ties breaking toward the lower-lift option. The three questions frame the sourced transition menu from the engagement's progress-report bundle; the scores are a reader-driven self-assessment rather than measured values, and the 313,000 USD/yr fixed-fee figure referenced elsewhere in the paper is an illustrative year-2 services number.

The matrix is deliberately a self-assessment, not a scorecard we filled in for the operator. The three questions are the ones the transition menu forces, and the tier your answers land on is the tier your ICT can actually run, which is a different and more useful thing than the tier with the best train-time. An operator that scores strong on talent and footprint and still reaches for the fixed fee is not wrong, but it should know it is buying speed it could have

produced itself, and paying an external dependency for the privilege. An operator that scores weak on both and reaches for in-house is making the expensive mistake this gate exists to prevent.

What the tiers do not change

It is worth naming what stays constant across all three tiers, because a clear-eyed decision needs the invariants as much as the variables.

The model is the same. Accuracy, the fracture and bedding picking quality, the dip and azimuth errors, none of it moves with the ownership tier. You are not trading accuracy for train-time; you are trading train-time and control against each other while accuracy sits fixed.

The data governance obligation is the same. In every tier the operator's wells and any personal data carry the same compliance requirements. The managed and in-house tiers keep the data on the operator's premises; the as-is tier runs on the provider's compute, which is a data-residency question the operator has to answer explicitly rather than an accuracy one. That answer, not the train-time, is often what rules a tier in or out first.

The residency question is worth sitting with, because it is where the cleanest technical option meets the hardest institutional constraint. Well data in a producing field is not casual data. It is contractually confidential, frequently covered by joint-venture and ministry obligations, and in the engagement we ran it fell under both wells-data and personal-data compliance regimes at once. The training pipeline handled this during R&D with a monthly signed reporting cadence and a data-management estate that kept versioned, access-controlled copies of every dataset. None of that governance disappears at handover; it moves to whichever tier owns the retrain loop. For the in-house and on-premises tiers the answer is straightforward, because the data never crosses the operator's boundary. For the as-is tier the operator has to be able to say, in writing and to its own auditors, that training its wells on a provider's external compute is permitted. Some operators can; some cannot; and the ones that cannot should discover it before they fall for the minutes-per-retrain number, not after.

And the retrain cadence is the operator's, not the tier's. None of the tiers force a retrain schedule. A cautious operator that retrains twice a year gets much less value from the minutes-versus-weeks difference than one that retrains every time a well lands. The value of the fast tier scales with how often you actually pull the trigger, which is a fact about the operator's workflow, not about the compute.

Sequencing the decision at the gate

Put the pieces in order and the ownership decision at the R&D-to-industrialization gate becomes a short, honest sequence rather than a reflex toward the fastest number.

Start with the data-residency answer, because it can eliminate a tier outright before any train-time comparison matters. If the operator's policy forbids training data leaving its premises, the as-is supercompute tier is off the table regardless of how fast it is, and the decision reduces to in-house versus managed on-premises. Answer the residency question first and you avoid falling in love with a tier you cannot use.

Then answer the three ICT questions honestly. Talent, footprint, commercial shape. Those three answers, more than any train-time figure, tell you which surviving tier your organisation can actually run for years.

Only then read the curve. With residency settled and the ICT questions answered, the train-time collapse from weeks to minutes is the tie-breaker and the sizing input, not the driver. It tells you what you gain by moving right along the curve, so you can decide whether the gain is worth the control and capability you give up to get it.

We put exactly this sequence in front of the operator we worked with, and the point of writing it down is that the sequence generalises. The specific hardware rungs and the specific fee band are ours; the shape of the decision, a train-time-versus-cost-and-control curve priced by questions about your own ICT, belongs to every operator standing at the same gate with a freshly-built model and a choice about who owns the machine that makes the next one.

WHAT THIS WHITEPAPER ARGUES

1. The real transition decision at the end of an R&D engagement is not about the model, which is portable and settled, but about who owns the compute that trains the next version. That ownership choice sets the operator's cost base, staffing, and retrain speed for years.
2. The three ownership models, replicate in-house, a light or medium managed service on your premises, and as-is managed supercompute, sit on one curve: per-model training time collapses from two-to-three weeks to minutes as you move right, while cost and internal-capability burden trade the other way. There is no dominating option, only a trade to price.
3. The train-time collapse is bought with compute that climbs three orders of magnitude, from a workstation GPU stack through a DGX A100 node (4-8x A100, up to 640 GB GPU memory, 2.5-5 petaFLOPS AI) to a 25-50 petaFLOPS SuperPod. Only the as-is fixed-fee tier puts the top rung under a retrain without a capital purchase, because the R&D engagement already paid the upfront investment.
4. Which tier fits is read off three questions about the operator's own ICT, whether it can hold GPU and MLOps talent for years, whether it has on-premises footprint and a staffable retrain SLA, and whether a fixed annual fee is the right budget shape, not off the fastest train-time on the slide.
5. Sequence the decision: answer data residency first because it can eliminate a tier outright, then answer the three ICT questions honestly, and only then read the curve as a tie-breaker and sizing input rather than the driver.

Limitations

This whitepaper generalises a single engagement's transition menu, and the generalisation has edges worth stating. The three tiers are the ones we actually offered a mid-sized carbonate operator; a very large operator with an existing high-performance-computing estate, or a very small one with no data centre at all, would see a different menu, and the middle managed tier in particular is sensitive to how much on-premises footprint already exists.

The train-time figures are documented as a weeks-to-days-to-minutes ladder across the three tiers, and we plot them on a logarithmic axis to show the collapse, but the exact minutes used to position each band on the chart are read from that ladder rather than being a benchmarked stopwatch number for a specific well and a specific retrain. Train-time in practice depends on dataset size, epoch count, and how much of the top-rung compute is actually allocated, and it will move with all three.

The compute-rung throughput values, the DGX A100 and SuperPod petaFLOPS figures, are sourced from the engagement's infrastructure records and the vendor system reference. The workstation rung's position on the throughput axis is a nominal placement, because the archive quotes that hardware by memory rather than by petaFLOPS, and it is marked as illustrative in the chart. The fixed-fee figure of 313,000 USD per year is an illustrative band from the year-two services frame, not a price list, and any real fee depends on scope, cadence, and contract term.

Finally, the ownership decision assumes the strategic question above it, whether to keep training in-house at all, is already settled. Operators that have not settled it should read the phased-operating-model and capability-handover work first; the ownership tiers here only make sense once the operator has decided it wants to own the retrain loop in some form.

References

Koroteev, D., Tekic, Z. (2021). Artificial intelligence in oil and gas upstream: Trends, challenges, and scenarios for the future. Energy and AI, 3, 100041. The upstream survey that frames why an operator would carry subsurface-AI compute in-house rather than rent it per project. https://www.sciencedirect.com/science/article/pii/S2666546820300033

NVIDIA (2020). NVIDIA DGX A100 System Architecture. White paper. The system reference for the DGX A100 node used as the mid-rung of the compute ladder. https://images.nvidia.com/aem-dam/Solutions/Data-Center/nvidia-dgx-a100-system-architecture-white-paper.pdf

Paleyes, A., Urma, R.-G., Lawrence, N. D. (2022). Challenges in Deploying Machine Learning: A Survey of Case Studies. ACM Computing Surveys, 55(6), 1-29. The evidence that the running cost of an ML system is dominated by what happens after the first model ships. https://dl.acm.org/doi/10.1145/3533378

EARTHSCAN

After the R&D Ends: Three Ownership Models for the AI You Just Built

Published October 2025. Generated 2026-07-22.
https://www.earthscan.io/whitepapers/three-ownership-models-for-the-ai-you-built