

The ML capability layer

Four progressive capabilities sit in the ML layer of the maturity model. They share one lifecycle (data preparation, development, deployment, operations), but they are not one capability. A churn model, a RAG system, an invoice-reprocessing agent and a real-time credit decision engine fail in completely different ways, and need completely different sign-off before launch.

This is a working reference, not a summary. The targets below are illustrative starting points; the questions are the useful part. Take the one capability you are actively investing in, answer its questions honestly, and fill in its current-state column. That filled-in gap, not the illustrative target, is what should drive your next prioritization conversation.

THE FOUR QUESTIONS BEHIND EVERY ROW

What is it? The outcome the enterprise can produce or protect, not the tools used to produce it.

How well must we do it? The performance expectation, and the gap between current and desired state.

Why does it matter? Which lever it pulls: revenue, cost, service, compliance, or future positioning.

What could go wrong? In performing it, or in failing to perform it at all.

The four capabilities at a glance

CAPABILITY	THE OUTCOME IT PRODUCES	WHERE ITS VALUE CONCENTRATES	THE QUESTION THAT EXPOSES YOUR GAP
1 · Predictive ML Models	A numeric or categorical outcome forecast from structured data, with every prediction traceable to the model version that made it.	The clearest, most direct revenue and cost case.	If this model degraded 15% next month, would anyone notice before a business user complained?
2 · Advanced ML Systems	Foundation-model and generative systems whose output we can trust, judged on grounding rather than accuracy.	Strategic differentiation, at some cost to immediacy.	How many of our POCs shipped with evaluation, prompt governance and monitoring?
3 · Process Automation	An operational workflow that used to need a person coordinating systems, now running end to end with defined error and escalation paths.	Efficiency and engineering capacity.	Which manual processes fail rarely enough, and safely enough, to automate first?
4 · ML-Driven Decisioning	ML output driving or supporting a real production decision, explainable and traceable to a model-plus-rules version.	Direct revenue and cost, carrying the highest stakes.	Can we explain and defend every decision this system made last quarter?

CAPABILITY 1

Predictive ML Models

The ability to forecast a numeric or categorical outcome from structured or semi-structured data (churn, demand, fraud scoring, anomaly detection), with every prediction traceable to the exact model version that produced it. Whether that runs on a built-in algorithm, a custom script or AutoML is an implementation choice below the capability line.

This is the category most stakeholders picture when they hear “machine learning,” and usually the one with the clearest value case, which also makes it the proving ground for the MLOps discipline everything else depends on.

Protect revenue	Demand forecasting and churn prediction let the business act before revenue is lost, not after.
Reduce cost	Fraud and anomaly detection catch losses earlier and cheaper than manual review; accurate forecasting cuts overstock and understaffing.
Improve service	Predictive maintenance and risk scoring shift the business from reactive to proactive.
Position for future	Registries, drift detection and retraining pipelines are built here first, and every other category inherits them.

PERFORMANCE DIMENSION	ILLUSTRATIVE TARGET	OUR CURRENT STATE
Accuracy / precision-recall, per use case	Meets or beats an agreed baseline: the current heuristic or legacy model	
Prediction traceability	100% of production predictions traceable to a registered model version	
Drift detection → retraining latency	Automatic retrain triggered within a defined window of detected drift	
Endpoint availability and latency	Meets the consuming application’s SLA, real-time or batch	

ASK BEFORE YOU SET A TARGET

- What are today’s predictions measured against: a documented baseline, or nothing at all?
- If this model degraded 15% next month, would anyone notice first? If not, the real gap isn’t accuracy; it’s the absence of automated drift detection.
- Do our feature pipelines match between training and serving? And who owns this model now that its builder has moved on?

CAPABILITY 2

Advanced ML Systems

The ability to build and operate foundation-model and generative systems (retrieval-augmented generation, customized or fine-tuned LLMs, multimodal models, and the agentic architectures layered on them), evaluated on grounding and output quality rather than a single accuracy score.

This capability introduces problems predictive ML never had to solve (chunking, embeddings, prompt governance), and is judged by an entirely different set of metrics.

Protect revenue	Product-embedded summarization, drafting and personalized recommendation can differentiate and retain.
Reduce cost	Deflects support and research labor by producing grounded first drafts instead of starting people from scratch.
Improve service	Fuller responses to complex or unstructured requests than a rules-based or purely predictive system can give.
Position for future	The technical foundation Process Automation and Decisioning build on whenever the workflow involves unstructured content or language reasoning.

PERFORMANCE DIMENSION	ILLUSTRATIVE TARGET	OUR CURRENT STATE
Groundedness / factual accuracy	≥ 90%, verified by an evaluation framework: model-as-judge, human review, or both	
Retrieval quality (RAG systems)	Top-k precision meets a threshold defined per use case	
POC-to-production conversion	Set locally	
Generation latency and cost per token	Meets defined SLA and unit-economics targets	

ASK BEFORE YOU SET A TARGET

- Not “can we build it” but: how many of our advanced-ML POCs shipped *with* governance, evaluation and monitoring, and how many are still running as unmonitored demos?
- How would we know the retrieval layer had drifted out of sync with its sources before a user did?
- Are prompts and fine-tuning configurations versioned artifacts? Can we explain why behavior changed after last week’s edit?

CAPABILITY 3

Process Automation

The ability to run operational workflows end to end with minimal manual intervention: document and data pipelines, agentic multi-step execution, and the automated retraining and deployment pipelines that cut ongoing engineering effort. This is an orchestration capability, not a single-model one: its unit of value is a completed workflow, not a prediction.

Reduce cost	Automates multi-step operational work (intake, exception handling, retraining) that consumed skilled hours per instance.
Improve service	Faster end-to-end turnaround on processes spanning several systems, with no waiting on manual handoffs.
Reduce cost (2°)	Cuts the engineering toil of manual retraining and redeployment, freeing capacity for new capability work.
Position for future	Builds the operational muscle (persistent state, branching and error logic, tested agent behavior) that Decisioning depends on.

PERFORMANCE DIMENSION	ILLUSTRATIVE TARGET	OUR CURRENT STATE
End-to-end completion rate	≥ 85% for the defined task class, without human rescue	
Tool-call / step failure rate	A defined ceiling, with automatic retry or fallback	
Time saved vs. the manual process	Set locally, per workflow	
Runaway loops / cost overruns	Zero tolerance, with hard stopping conditions enforced	

ASK BEFORE YOU SET A TARGET

- Which processes are candidates for automation at all, and which stay human-led regardless of technical feasibility? That is a governance decision, not a technical one.
- Of the processes run manually end to end today, which fail rarely enough, and safely enough, to automate first?
- Would we notice a workflow that silently stalled, or duplicated a payment or a communication, before a downstream complaint surfaced?

CAPABILITY 4

ML-Driven Decisioning

The ability to have ML output directly drive or support a business decision in production (approval scoring, recommendation ranking, dynamic pricing, risk decisioning, human-in-the-loop review), with every decision explainable and traceable to a specific model-plus-rules version.

This is the highest-stakes capability in the layer. The outcome isn't a prediction; it is an action taken about a real customer, applicant or transaction, often in a regulated context.

Protect revenue	Real-time pricing and recommendation ranking shape transaction value and conversion directly.
Reduce cost	Automated approval and risk decisioning removes manual underwriting or review for the straightforward majority.
Achieve compliance	Explainability, bias monitoring and audit trails are frequently the specific controls a regulator asks to see evidenced.
Avoid risk cost	Prevents the cost of a biased or unexplainable decision reaching a regulator or a court, far larger than building the governance in from the start.

PERFORMANCE DIMENSION	ILLUSTRATIVE TARGET	OUR CURRENT STATE
Explainability coverage	100% of production decisions carry feature attribution and confidence scoring	
Bias / fairness monitoring	Continuous monitoring against defined protected-class thresholds	
Human-in-the-loop escalation rate	Tracked, trending to the rate that fits the decision's risk tier, not zero; zero usually means loose guardrails	
Shadow validation before cutover	100% of new decision models shadow-tested against live traffic before promotion	

ASK BEFORE YOU SET A TARGET

- Can we explain and defend every decision this system made in the last quarter? A no is a governance gap, whatever the accuracy metrics say.
- Are the rules layer and the model kept in step, or can one be updated without the other?
- Who has sign-off authority before a new decisioning model launches, and has anything reached production without that review?

Four things that hold across all four

Value concentrates unevenly, so a revenue-only filter misprices the middle.

Predictive ML Models and ML-Driven Decisioning carry the clearest direct revenue and cost cases. Advanced ML Systems trades some immediacy for strategic differentiation; Process Automation is mostly efficiency and capacity. Prioritize on “biggest revenue case” alone and you underfund the orchestration and governance work that keeps the other three reliable and accountable at scale.

Risk changes in kind, not just in degree.

Predictive ML’s risks are mostly about silent degradation. By Decisioning they are about explainability, fairness and regulatory exposure to real people. One evaluation framework applied uniformly across all four will over-engineer the low-stakes categories and under-govern the high-stakes ones.

Same lifecycle, different finish lines.

“Done” for a predictive model means a versioned, monitored endpoint. “Done” for a decisioning system means that plus explainability, bias monitoring and audit sign-off. Applying the predictive definition of done to a decisioning system is a common and costly mistake.

A gap is only as good as the current-state number behind it.

Every illustrative target here needs a real current-state figure before it becomes actionable. The fastest route to usefulness is to fill in the current-state column for the one or two capabilities under active investment this planning cycle, not to complete all four at once.

DO THIS NEXT

Pick the single capability under active investment or debate right now. Complete its performance table with real current-state numbers from whatever monitoring exists today, even informal monitoring. That filled-in gap, not the illustrative targets above, is what should drive the next round of prioritization and investment conversations.