SUPPLY CHAIN RESEARCH · EDITORIAL FEATURE

The Data Quality Reckoning

Why AI and Advanced Planning Fail on the Foundation Nobody Funds

For decades, bad data was absorbed by humans who knew the number looked wrong. Automation removes that check, and the models now execute the errors instead of catching them. This is what AI-ready data actually requires, and why the reckoning has arrived.

Independent, vendor-neutral research for supply chain and technology leaders

Supply Chain Research | supplychainresearch.com | 2026

Contents

A decision framework for the AI era

01 Executive summary

Every organization now buying artificial intelligence for its supply chain is being sold a capability that depends on a foundation it has not built. The models are real, the vendors are largely competent, and the use cases are sound. What is missing is the data underneath: master records that reconcile, inventory positions that match the shelf, product definitions that mean the same thing in every system, and a governance discipline that keeps them true. Without that foundation the model does not fail loudly. It succeeds, confidently, on the wrong inputs, and it produces a plausible answer that nobody can distinguish from a correct one.

For decades this did not matter as much as it should have, and the reason is worth understanding, because it explains why the bill has come due now rather than twenty years ago. Bad data was absorbed by people. A planner looked at a forecast, knew from experience that the number was wrong, and adjusted it. A buyer saw an inventory position that did not match what they remembered walking past in the warehouse, and went to check. A finance analyst spotted a total that could not be right and traced it back. Human judgment was a quiet, expensive, and remarkably effective error-correction layer sitting on top of data that was never as good as anyone claimed. Automation removes that layer. A model does not know the number looks wrong, and an agent that acts on the model does not pause to check. We say honestly that the data was always this bad. What has changed is that the organization has stopped catching it.

~73%
of enterprise data leaders rank data quality as the primary barrier to AI success, ahead of models, compute, and talent

~65%
estimated inventory record accuracy in the average retail store, against a world-class benchmark of 95 percent

~16%
of master data management programs funded as enterprise-wide strategic initiatives

The argument in brief

- **The data was always this bad.** Inventory accuracy, forecast accuracy, and master data consistency have been mediocre for decades, and human judgment quietly compensated.
- **Automation removes the human check.** A model trained on bad data learns the error; an agent acting on the model executes it. The error becomes a transaction rather than a conversation.
- **Master data is the substrate.** Most data quality failures are not typographical but semantic: the same entity means different things in different systems, and no model can infer its way past that.
- **The economics are brutal and well understood.** It costs roughly one unit to verify a record at entry, ten to cleanse it later, and one hundred to carry it through the decisions it corrupts.
- **The foundation is fundable.** Data quality is not a technology problem awaiting a better tool. It is an ownership problem awaiting an owner, and the organizations that assign one solve it.

02 The reckoning: why now and not before

The complaint that enterprise data is poor is not new. It has been made, accurately, in every decade since organizations began keeping records in computers, and it has been made so often and acted on so rarely that it has acquired the character of weather: a permanent condition to be endured rather than a problem to be solved. What has changed is not the weather. It is that the organization has taken off its coat.

The mechanism is set out in Figure 1, and it is the single most important idea in this article. Consider what happened, historically, to a bad record. It entered a system, it flowed into a report, and a human being looked at the report. That human being possessed something no system possessed: a model of what the number ought to be. The planner knew that this product does not sell four thousand units in a week. The warehouse supervisor knew that pallet was not there. The controller knew that margin could not be correct. The bad record was caught, not because any system caught it, but because a person with context found it implausible and went to look. That quiet, distributed, wildly inefficient error-correction process was the reason organizations could run on data that was, by any honest measurement, substantially wrong.

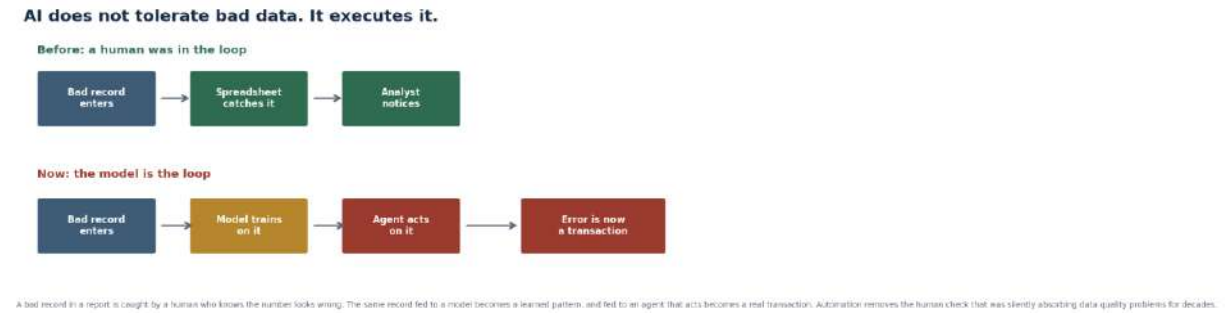

Figure 1. What automation removes. The human who caught the implausible number was the error-correction layer, and models do not replace that function. They execute what they are given.

Now consider what happens to the same bad record in an automated pipeline. It enters a system, it flows into a training set, and a model learns from it. The model does not find it implausible, because the model has no independent notion of what is plausible; it has only the data, and the data says this is what happens. If the record is one of many, the model learns a slightly wrong pattern. If the errors are systematic, and errors in enterprise data usually are, the model learns a confidently wrong pattern. And then an agent, downstream of the model, acts on it: it reorders, it reallocates, it commits a delivery date, it releases a purchase order. The error is no longer a number on a report that somebody might question. It is a transaction that has already occurred.

This is why the reckoning has arrived now. The organizations racing to deploy artificial intelligence across their supply chains are, in effect, removing the only quality-control mechanism their data has ever had, and replacing it with a system that is faster, more consistent, and entirely incapable of noticing that its inputs are wrong. The technology is not the problem. The technology is doing precisely what it was asked to do, at scale, on a foundation that was never good enough to bear it.

There is a second reason the reckoning arrives now, and it compounds the first. Automation does not merely remove the human check; it also increases the volume and speed of decisions to a level at which human checking would be impossible even if the organization wanted it. A planner reviewing a hundred replenishment recommendations a day can bring judgment to each. A system generating a hundred thousand replenishment decisions a day cannot be reviewed by anyone, and the organization that deploys it has, whether or not it acknowledges the fact, made an irrevocable decision to trust its data. That is a perfectly reasonable decision if the data merits the trust. It is a catastrophic one if the data is right two-thirds of the time, and almost nobody making the decision has measured which situation they are in.

The historical pattern also explains why the warnings were ignored for so long, and why they should not be ignored now. For thirty years, data quality specialists have said that enterprise data was unfit for the purposes it was being put to, and for thirty years the organizations that ignored them were, in a narrow sense, right to do so, because the systems in question degraded gracefully. A wrong number in a report produced a slightly worse decision, which a competent manager partially corrected. The cost was real and diffuse and never large enough in any single instance to force action. What has changed is that automated systems do not degrade gracefully. They fail silently and confidently, at scale, and the diffuse cost becomes a concentrated one. The warnings were always accurate. They were merely, until now, ignorable.

A further consequence of the shift deserves attention, because it changes what an organization must be able to prove. When a human made the decision, the organization could explain it: the planner overrode the forecast because they knew a competitor was on promotion. When a model makes the decision, the explanation is the data, which means that an organization which cannot vouch for its data cannot explain its own decisions. That is uncomfortable in a commercial setting and it is untenable in a regulated one, and it will become a governance issue for boards well before it becomes a technology issue for engineers. The lineage of a number, where it came from, who owns it, when it was last verified, ceases to be a technical nicety the moment a machine is acting on it and a regulator or a customer asks why.

The counterargument that some vendors will offer deserves an honest hearing, because it is not entirely without merit. It is claimed that modern models are robust to noise, that they can learn around errors given sufficient volume, and that waiting for perfect data is a counsel of paralysis. The first part is partly true: models do tolerate random noise reasonably well, and an organization waiting for pristine data will wait forever. But the errors in enterprise data are overwhelmingly not random. They are systematic, which is the one kind of error that models cannot learn around, because a systematic error looks exactly like a real pattern. A duplicate customer is not noise; it is a consistent misstatement that the model will faithfully learn. The vendors are right that perfection is not required. They are wrong, and sometimes knowingly wrong, that the errors that actually exist are the kind that models absorb.

03 Handle the headline number with care

Anyone who reads about this subject will encounter, within a few paragraphs, the claim that poor data quality costs the average organization twelve point nine million dollars a year. It appears in vendor decks, in board papers, in conference keynotes, and in almost every article written on the topic. Because this publication is vendor-neutral and because a buyer's credibility depends on citing figures that survive scrutiny, it is worth being precise about what that number is and what it is not. Figure 2 sets out the provenance.

Figure 2. The most-quoted figure in the field, and what it actually measures. Cite it with its provenance or do not cite it.

The figure originates in a vendor-landscape study published in 2020. As part of that research, the analyst firm surveyed roughly one hundred fifty reference customers of data quality software vendors and asked them to estimate what poor data quality was costing their organizations. The average of those estimates was twelve point nine million dollars. Every element of that sentence matters. It is an estimate, not a measurement. It is self-reported by the organizations themselves. The sample consists of large enterprises that were already sophisticated enough to be purchasing data quality software, which is a self-selected population that has both thought about the problem and has a reason to believe it is expensive. And it is from 2020.

None of this makes the figure worthless. As an order of magnitude for a large enterprise it is probably about right, and it is useful precisely because it converts an abstract complaint into a number a chief financial officer can react to. But it is not an audited measurement, it does not scale to organizations of other sizes, and a buyer who quotes it as though it were a hard finding will be embarrassed by anyone who checks. The correct handling is to cite it with its provenance attached, or to reach instead for the figures that are better grounded: the finding from academic research that poor data quality costs organizations somewhere in the range of fifteen to twenty-five percent of revenue, the analyst finding that a majority of generative artificial intelligence projects are abandoned after the proof of concept with poor data quality among the leading causes, or simply the organization's own measurement of its own data, which is the only figure that will ever actually persuade anyone to act.

This point generalizes, and it is worth stating as a principle rather than a footnote. The data quality field is unusually full of large, round, unattributable numbers, many of them tracing back to a single study, a single interview, or a single vendor's marketing department, and repeated so often that they acquire the appearance of established fact. A leader building a case for investment in this area should assume that

any striking statistic requires verification, should trace each one to its source before repeating it, and should recognize that the most persuasive number available is never the industry average. It is the measurement of the organization's own inventory accuracy, taken last month, in its own warehouse.

A related and equally useful discipline is to interrogate the denominator whenever a data quality claim is made. A statement that data quality costs the average organization some enormous sum is unanswerable, because average of what, measured how, across which industries and which sizes. A statement that this organization's inventory accuracy in this distribution center was measured last month at seventy-four percent against a physical count is answerable, actionable, and impossible to wave away. The first kind of number produces nodding. The second kind produces budgets. A leader who wants to move an organization on this issue should spend less time collecting industry statistics and more time commissioning a count.

It is worth naming the mistake that the headline figure encourages, because it is the reason so many data quality business cases fail even when the underlying problem is severe. A leader who walks into a budget meeting armed with an industry average is making an argument that can be deflected in one sentence: that is the industry, and we are better than the industry. The deflection is almost always wrong, and it is unanswerable, because the leader has no evidence about this company. The alternative argument, that a physical count of this warehouse last month found the system wrong on nearly a third of the items sampled, cannot be deflected at all. It converts a debate about statistics into a conversation about a fact, and the conversation ends with somebody asking what it would cost to fix.

The provenance discipline recommended here has a wider application that is worth stating, because it will serve a leader well across every technology decision. Any striking number presented in support of a purchase should be traced to its source before it is repeated, and the trace should establish four things: who measured it, what population they measured, when, and whether they had an interest in the result. Most numbers in enterprise technology marketing fail at least one of those tests, and many fail all four. The exercise takes minutes, it is almost never performed, and it is the cheapest form of diligence available. An organization whose leaders habitually perform it will make better decisions than one whose leaders repeat whatever appeared on the slide, and the difference compounds over every decision either of them makes.

04 What the data in a supply chain actually looks like

Abstractions about data quality are easy to nod at and easy to ignore. The specific numbers are harder to ignore, and in supply chain they are available, because the discipline has the useful property that its data describes physical objects which can be counted. The comparison between what the system says and what is actually on the shelf is the most honest data quality audit any organization can perform, and the results, shown in Figure 3, are sobering.

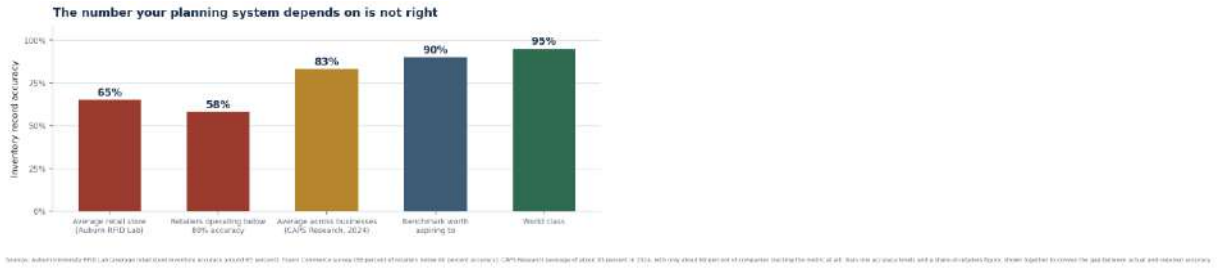

Figure 3. Inventory record accuracy against the benchmark. The number that every planning system, every replenishment decision, and every promise to a customer depends on.

Research from a university laboratory that studies this specifically has put the average inventory accuracy of a retail store at around sixty-five percent, which is to say that roughly one item in three is not where the system believes it to be, or is not there at all, or is there in a quantity the system does not know about. A survey of retailers found that a majority operate below eighty percent accuracy. Broader benchmarking across industries puts the average at around eighty-three percent, which is better, and notes something equally troubling: only about two-thirds of organizations track the metric at all. The benchmark that supply chain professionals consider worth aspiring to is ninety percent, and world-class operations achieve ninety-five.

Sit with the implication of those numbers for a moment, because it is easy to read them as a technical statistic rather than as what they are. Every replenishment decision, every promise to a customer about availability, every safety stock calculation, every forecast, and every planning run is computed from an inventory position. If that position is right sixty-five or eighty percent of the time, then the sophisticated optimization engine sitting on top of it is optimizing a fiction with great precision. The organization has bought a system that computes the correct answer to the wrong question, and the answer arrives with all the confidence that an algorithm produces and none of the doubt that a planner would have brought.

Forecast accuracy tells a similar story, shown in Figure 4. Average forecast accuracy in consumer packaged goods runs at around fifty-five percent, and around seventy percent among technology firms, and research finds that a majority of manufacturers report that inaccurate demand forecasts drive more than fifteen percent excess inventory every year. Some of that inaccuracy is irreducible, because demand is truly uncertain and no model can predict what has not yet been decided by customers who have not yet decided it. But a substantial share of it is not irreducible at all: it is the downstream consequence of feeding a forecasting model sales history that is polluted by stockouts recorded as zero demand, by promotional periods that were never flagged, by product hierarchies that changed without anyone restating the history, and by an inventory position that was never right in the first place.

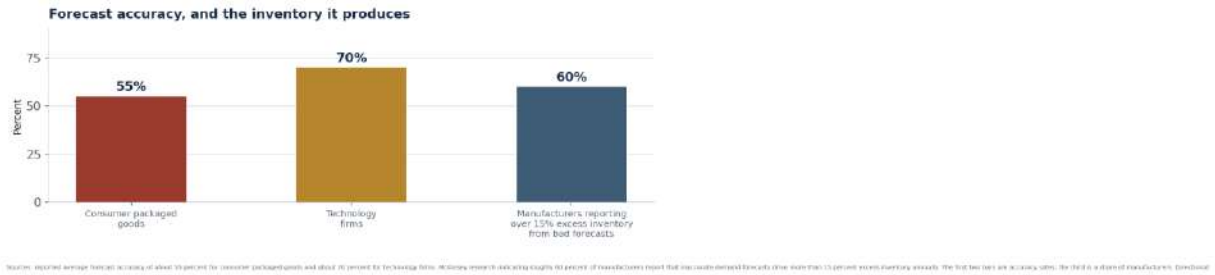

Figure 4. Forecast accuracy and the excess inventory it produces. Some inaccuracy is irreducible; a substantial share is a data problem wearing a forecasting costume.

The uncomfortable conclusion, and the one that should govern how an organization allocates its next dollar of supply chain investment, is that a large share of what is diagnosed as a forecasting problem, a planning problem, or an optimization problem is in fact a data problem presenting under a different name. Organizations respond by buying better algorithms, because algorithms are purchasable and data quality is not. The algorithm arrives, it is fed the same data, and it produces a marginally different wrong answer, at which point the organization concludes that the vendor overpromised. Sometimes the vendor did. Often the vendor delivered exactly what was sold, into an environment that could not support it.

Sales history deserves particular attention as a data source, because it is the input on which almost every demand model depends and it is almost universally corrupted in ways that nobody corrects. Consider what a sales record actually captures: it captures what was sold, which is not the same as what was demanded. A week in which a product was out of stock records low sales and is read by the model as low demand, so the model forecasts low demand, so the system stocks less, so the product goes out of stock again. This is a self-reinforcing error, it is extremely common, and it is invisible unless the organization has recorded its stockouts, which most do not. Every model trained on uncorrected sales history learns to under-forecast exactly the products that most need forecasting well.

The same corruption enters through promotions, price changes, and product hierarchy restructuring. A promotional week produces a spike that the model, unless told, treats as underlying demand. A price change alters the demand curve in ways that the history cannot explain unless the price is a feature. A hierarchy restructuring, where products are recategorized, silently rewrites the history of every category above them, and the model, which has no memory of the old structure, learns from a past that has been retroactively edited. None of these is an error in the sense of a wrong number. Every record is accurate. The data is nonetheless unfit for the purpose, because it does not carry the context that would make it interpretable, and a model cannot supply context it was never given.

Two further characteristics of supply chain data make it harder than the enterprise average, and both are worth understanding because they mean that benchmarks drawn from other functions understate the difficulty. The first is that a great deal of supply chain data originates outside the organization: supplier lead times, carrier transit times, customer forecasts, and trading partner catalogs are created by counterparties, on their own systems, to their own standards, and arrive in whatever condition they arrive in. The organization cannot impose entry validation on a supplier. The second is that supply chain data describes physical objects that move, which means it goes stale by the hour rather than by the

year, and a position that was accurate this morning may be wrong by lunchtime. Together these mean that a supply chain cannot solve its data problem purely by tightening its own processes; it must also decide how much to trust what arrives from outside, and design for the answer.

This external dimension deserves a specific practice, because it is routinely neglected. Data arriving from a counterparty should be validated at the boundary, before it enters the organization's systems, exactly as data entered by an employee should be validated at the screen. A supplier catalog with missing units of measure should be rejected at the door, with the exception routed back to the supplier, rather than accepted and cleaned up internally, which is how organizations end up permanently subsidizing their suppliers' data quality with their own staff. The boundary is the cheapest place to enforce a standard and the only place where the cost lands on the party that can actually fix the problem. Organizations that enforce it find, to their surprise, that suppliers comply.

05 Why AI amplifies bad data rather than absorbing it

The claim that data quality is the primary obstacle to artificial intelligence is not a claim this publication is making on its own authority. It is what the people running these programs report, shown in Figure 5. Surveys of enterprise data leaders find that roughly three-quarters rank data quality as the leading barrier to success with artificial intelligence, ahead of model accuracy, ahead of compute cost, and ahead of the shortage of talent that receives far more attention. Analyst research finds that a majority of generative artificial intelligence projects are abandoned after the proof of concept, with poor data quality among the top reasons cited. And the widely discussed finding that the overwhelming majority of enterprise pilots produce no measurable financial impact is, when the failures are examined, substantially a story about systems that could not get clean, contextual, reliable data.

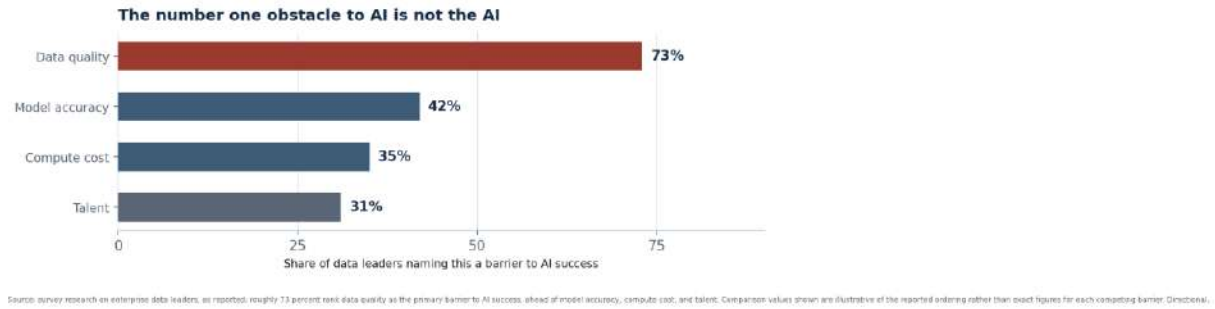

Figure 5. What the people running these programs say is stopping them. The obstacle is not the model.

The mechanism of amplification is worth spelling out, because understanding it is what turns a vague anxiety into a specific plan. A traditional report is a passive artifact: it displays a number and waits for a human to interpret it. A model is an active one: it extracts a pattern from the data and then applies that pattern to new cases. If the data contains a systematic error, and enterprise data almost always does, the model does not merely repeat the error; it generalizes it. A duplicate customer record does not just produce one wrong report; it teaches the model that this customer buys half as much as they do, and the model then applies that belief to every forecast, every recommendation, and every allocation decision involving that customer, forever, until somebody notices.

Autonomy compounds this further, and this is the part that should concern a supply chain leader most. A recommendation engine that produces a wrong recommendation is checked by the person who receives it, and that person, being a planner with context, may well reject it. An agent authorized to act does not present its conclusion for review; it executes. The bad data becomes a purchase order, a reallocation, a commitment to a customer, or a cancelled shipment, and it does so at machine speed and machine volume, which means the organization can accumulate a great many such errors in the time it would previously have taken to notice one. The move from advice to action raises the required data quality bar by an order of magnitude, and almost no organization deploying agents has raised its data quality by any amount at all.

It follows that the honest sequencing advice, which many vendors will not offer because it delays a sale, is that the data work comes first. An organization whose inventory accuracy is sixty-five percent should not be buying an autonomous replenishment agent; it should be fixing its inventory accuracy, and it will

find, when it does, that its existing systems perform substantially better than they did, which is a return it can bank before it spends anything on artificial intelligence at all. This is not an argument against the technology. It is an argument for the order of operations, and the order of operations is not negotiable, because a model cannot infer a fact that its data does not contain.

The correct response to all of this is not despair and not a moratorium on artificial intelligence. It is a change in what the organization measures before it decides. Any serious evaluation of an artificial intelligence capability in supply chain should begin with a data readiness assessment, conducted candidly, and it should be permitted to conclude that the organization is not ready, which is a conclusion that almost no evaluation is currently structured to reach. A vendor evaluation that can only end in the selection of a vendor is not an evaluation. The most valuable outcome of a well-run assessment is frequently the discovery that the twelve months and the budget earmarked for a model would produce a far larger return if spent on the inventory accuracy that the model would have depended on.

The order-of-operations point has a corollary that is worth stating for anyone currently under pressure to show artificial intelligence progress, because the pressure is real and this article should not pretend otherwise. The right response to that pressure is not to refuse, which will simply mean that somebody else runs the program without the data caveats. It is to insist that the pilot be scoped where the data can actually support it, which is usually a narrower domain than the ambition, and to make the data readiness of the wider domain an explicit output of the pilot rather than an assumption behind it. A pilot that succeeds in a bounded area and produces a credible, evidenced account of what the data would need to be for the capability to scale has done far more for the organization than one that attempts the full ambition and fails for reasons nobody diagnoses.

A short note on where the responsibility for this actually sits, because the temptation is to place it on the vendor and the temptation should be resisted. A vendor selling a demand planning system is not obliged to fix the buyer's sales history, and a vendor selling an autonomous replenishment agent is not obliged to count the buyer's warehouse. What a vendor is obliged to do, and what the better ones do, is state clearly what the system assumes and what it will do when the assumption fails. A buyer who does not ask that question has not been misled; they have failed to ask. The distribution of responsibility here is uncomfortable but fair: the vendor owes candor about requirements, and the buyer owes a candid assessment of whether it can meet them, and the failures documented throughout this article are overwhelmingly failures of the second obligation rather than the first.

06 What AI-ready data actually means

The phrase AI-ready data is now used so loosely that it has stopped conveying anything, which is a pity, because the concept behind it is precise and useful. Data is ready for automation when it satisfies a set of conditions that build on one another, shown in Figure 6, and an organization can locate itself candidly on that ladder in an afternoon.

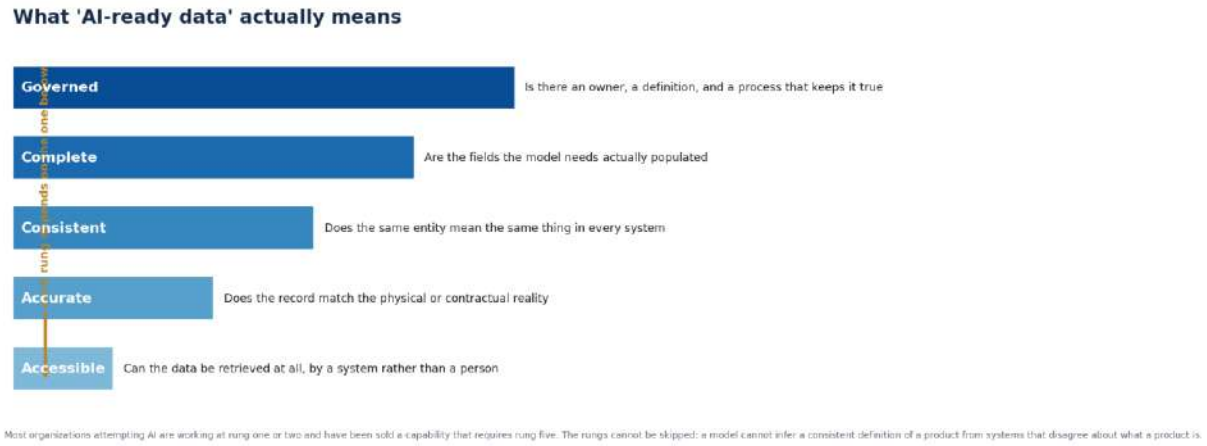

Figure 6. The ladder of data readiness. The rungs cannot be skipped, because each depends on the one beneath it.

- **Accessible.** Can the data be retrieved programmatically, at the frequency the use case requires, by a system rather than by a person exporting a spreadsheet? A great deal of enterprise data fails at this first rung, trapped inside applications that will not release it cleanly.
- **Accurate.** Does the record correspond to reality, to the pallet on the rack and the contract in the file? This is the rung that inventory accuracy measures, and the one on which most supply chain organizations sit at somewhere between sixty-five and eighty-five percent.
- **Consistent.** Does the same entity mean the same thing across every system? This is the master data rung, and it is the one that most decisively determines whether an artificial intelligence program will work, because a model cannot reconcile definitions that the organization itself has never reconciled.
- **Complete.** Are the fields the model actually needs populated, for enough of the records, across enough of the history? A field that is optional in the system and empty in eighty percent of records is a feature the model cannot use, however important the business believes it to be.
- **Governed.** Is there a named owner, an agreed definition, a measured quality level, and a process that keeps the data true as the business changes? Without this rung, every improvement made at the rungs below decays, because data quality is not a state that is achieved but a condition that is maintained.

The value of the ladder is diagnostic. An organization that places itself candidly on it will usually discover that it is operating at the second rung and has been sold a capability that requires the fifth, which explains, without any need for further analysis, why the pilot did not work. And the ladder makes clear why the fashionable interventions do not help: a better model does not fix inconsistency, more compute does not fix incompleteness, and a more expensive platform does not fix the absence of an owner. Each rung must be climbed, in order, and there is no product that climbs them for you.

It is worth stating what happens at each rung when it is not satisfied, because the failure modes are distinct and are frequently misattributed. A failure at the accessible rung looks like a project that never starts, stalled in an argument about extracts and permissions. A failure at the accurate rung looks like a model that performs beautifully in testing on historical data and poorly in production, because the historical data was reconciled and the live data is not. A failure at the consistent rung looks like a model that works in one business unit and not in another, because the entities mean different things in each. A failure at the complete rung looks like a model that ignores the feature the business considers most important, because the field was empty. And a failure at the governed rung looks like a model that worked for six months and then degraded, because the data drifted and nobody was watching. Each of these is routinely diagnosed as a modeling problem. None of them is.

It is worth being concrete about what the fifth rung, governance, actually consists of, because the word is used so loosely that it has become a synonym for meetings. Governance here means four specific things and nothing more. A definition: what this field means, written down, agreed. An owner: a named person, in the business, accountable for it. A measurement: the current quality level, established against reality and published where peers can see it. And a process: what happens when the measurement falls, and who is responsible for making it rise. Four items, per critical field. An organization that has these for its twenty most important fields has better data governance than one with an elaborate council, a data catalog, and a policy document that nobody has read.

The ladder also provides the honest answer to a question that boards increasingly ask, which is how long the foundation work takes. The answer depends on which rung the organization starts from, and it can be estimated with reasonable confidence. Getting data accessible is a matter of months and is largely an engineering task. Getting it accurate, for a defined scope, is a matter of a quarter or two and is largely an operational one, requiring counting and correction. Getting it consistent, which is the master data rung, is the long one, typically a year or more for a meaningful domain, because it requires organizational decisions that cannot be rushed and that will be resisted. Getting it complete follows quickly once consistency is established. And governance is not a phase at all; it is the state the organization enters and does not leave. A leader who presents that timeline candidly will be trusted. One who promises the foundation in a quarter will not be, twice.

07 Master data: the foundation nobody funds

The third rung deserves its own section, because it is the one that organizations most consistently misdiagnose. When people say data quality they usually picture errors: a typo, a wrong number, a missing field. Those exist, and they matter, and they are the easy part. The hard part, and the part that actually defeats artificial intelligence programs, is not that records are wrong but that they are inconsistent, which is a different failure with a different remedy.

Master data is the definitive record of the entities the whole business depends on: the customer, the product, the supplier, the location, the asset. In a fragmented estate these records diverge, and they diverge in ways that are individually reasonable and collectively fatal. The same customer exists as three records because it was entered by three teams with three conventions. The same product carries one code in the enterprise system, another in the warehouse system, and a third in the e-commerce catalog, in different units of measure, under a hierarchy that was restructured two years ago without anyone restating the history. A supplier is a legal entity in the procurement system and a shipping address in the logistics system, and no field links them. None of these records is wrong. Each is locally correct and globally incoherent, and no model, however capable, can infer the mapping that the organization itself has never made.

The research on how organizations manage this, shown in Figure 7, is not encouraging. Only a small minority of master data programs are funded as enterprise-wide strategic initiatives. A clear majority of organizations have no well-defined process for integrating new data sources with the ones they already have. Fewer than a third have master data integrated fully both upstream and downstream. And a striking majority of organizations lose a day or more every week to resolving master data quality problems, which is to say that the cost of not solving the problem is already being paid, in the most expensive possible currency, as the unmeasured time of skilled people doing reconciliation by hand.

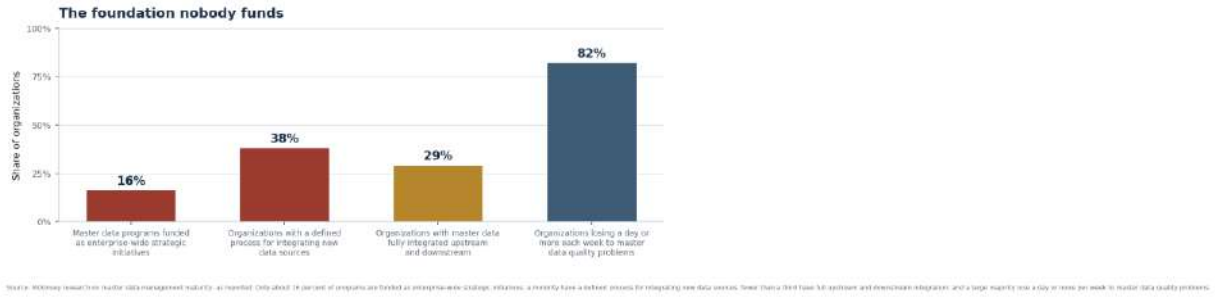

Figure 7. Master data management maturity. The discipline that determines whether every other data initiative succeeds is the one least likely to be funded as a strategic program.

Why is it so consistently underfunded? Because it produces nothing visible. A master data program does not ship a feature, does not appear on a dashboard, and cannot be demonstrated to a board in a way that produces applause. Its entire return consists of other things working, which is a return that is real and is nearly impossible to attribute. So the money goes instead to the visible thing, the planning system or the artificial intelligence pilot, which is then deployed onto the unreconciled foundation and fails for reasons that get attributed to the vendor. Analysts have been cited as estimating that around three-quarters of master data management programs fail to meet their objectives, and a leader should read

that figure not as evidence that the discipline does not work but as evidence that it is habitually attempted without the mandate, the ownership, or the funding it requires.

There is a reason the semantic problem is harder than the typographical one, and it is worth naming because it explains why technology alone will not solve it. A typo is a deviation from a known correct value, and software can therefore detect it: a postal code with a letter where a digit belongs is detectably wrong. But when two systems hold different definitions of a product, neither is wrong. Each is correct within its own context, and there is no external truth against which software can adjudicate, because the truth is a decision the organization has never made. Somebody has to decide what a product is, which system owns that decision, and how every other system will conform to it. That is an act of governance, not of engineering, and no amount of technology substitutes for it. This is why master data programs cannot be delegated to the technology function and why the ones that are delegated there reliably fail.

The three pathologies that most damage artificial intelligence deserve to be named together, because they attack a model in a way that reports never revealed. Duplication teaches the model that one entity is several, which fragments its view of demand and understates the importance of the organization's largest customers. Inconsistency prevents the model from joining the data at all, so it either fails or, worse, joins it incorrectly and produces plausible nonsense. And incompleteness starves the model of the feature that carries the signal, so it learns from what is present rather than from what matters. A dashboard tolerates all three, because a human reading the dashboard supplies the missing understanding. A model does not read; it fits, and it will fit whatever it is given.

None of this argues that the rungs must be perfect before anything can be attempted, and it would be a misreading of this article to conclude so. Perfection is neither achievable nor necessary; the useful question is not whether the data is clean but whether it is good enough for the specific decision being automated, and the answer varies enormously by use case. A model that recommends which of two hundred exceptions a human should look at first can tolerate substantial noise, because a human still decides. A system that autonomously commits a delivery date to a customer cannot. The readiness bar is set by the consequence of being wrong and by whether anyone is checking, which means the same data can be adequate for one application and dangerous for another, and an organization that assesses readiness use case by use case will find far more it can do than one that treats data readiness as a single gate.

The ladder repays one more use, as a filter on the vendor conversation. When a vendor demonstrates a capability, the buyer should ask which rung the demonstration assumed. The answer, if the vendor is candid, is usually the fifth: the demonstration data was accessible, accurate, consistent, complete, and governed, because the vendor prepared it. The buyer's data is at the second rung. The gap between those two positions is the entire implementation risk of the program, and it is a gap the vendor cannot close, because it lies inside the buyer's organization and consists of decisions only the buyer can make. Naming that gap explicitly, in the evaluation, converts a vague sense of unease into a scoped workstream with a cost and a timeline, which is the only form in which it will ever get funded.

None of the five pathologies is fully solvable by software, and it is worth being blunt about why, because a great deal of money is spent on the contrary assumption. Software can detect a duplicate, but it cannot decide which of the two records is correct, because that requires knowing the business. Software can detect that two systems disagree, but it cannot decide which one should win, because that is a question of authority rather than of fact. Software can flag an empty field, but it cannot fill it. In every case the tool surfaces the problem and a human resolves it, which means the binding constraint on data quality is not detection but resolution capacity, and resolution capacity is a function of how many people are accountable and how much authority they have. Organizations buy detection and starve resolution, and then wonder why the reports pile up unread.

08 The economics: one, ten, one hundred

There is a rule of thumb in data quality practice that deserves to be better known outside it, because it converts an argument about diligence into an argument about money, which is the only argument that reliably moves a budget. It is called the one-ten-one-hundred rule, and it is shown in Figure 8. It costs roughly one unit of effort to verify a record at the moment it enters the organization. It costs roughly ten units to find and cleanse that same record later, in a remediation project. And it costs roughly one hundred units to live with it: to carry the bad record through every system it propagates into and every decision it corrupts, and to pay for the consequences.

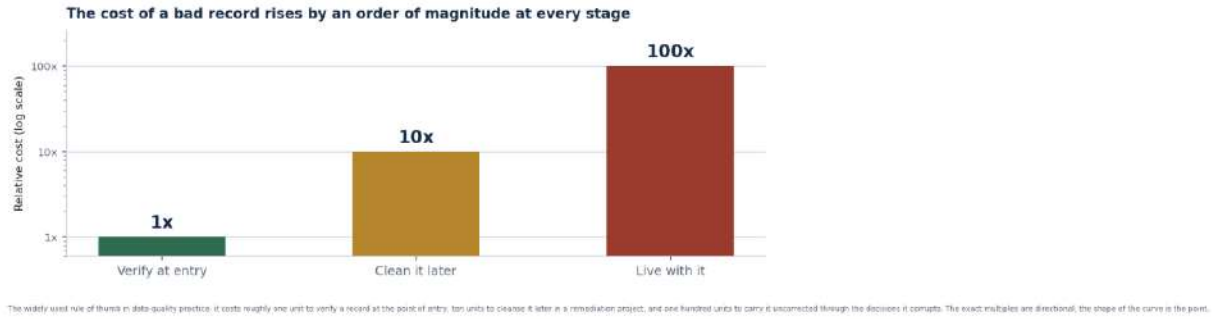

Figure 8. The one-ten-one-hundred rule. The exact multiples are directional; the order of magnitude at each step is the point.

The exact multiples are directional and should be presented as such, but the shape of the curve is not in dispute and is intuitively obvious once stated. A bad record caught at entry costs a validation rule and a moment of somebody's attention. The same record caught two years later must be found, which requires an investigation, and corrected, which requires understanding what it should have been, and then reconciled across every downstream system it has since flowed into, each of which may have made its own adjustments. And the same record never caught at all sits in the data forever, producing wrong forecasts, wrong allocations, wrong customer promises, and wrong decisions, and the cost of those is unbounded because nobody ever counts them.

The strategic implication is direct and slightly counterintuitive. The highest-return investment in data quality is almost never a cleansing project, which is the ten, and it is certainly not a better analytics platform, which does nothing about any of this. It is validation at the point of entry, which is the one: the rule that will not let a product be created without a unit of measure, the check that prevents a duplicate customer at the moment of creation, the requirement that a supplier record carry a tax identifier before it can be used. These controls are unglamorous, they irritate the people who have to comply with them, and they are the only intervention in this entire field whose cost is a fraction of the problem it prevents.

This also explains why remediation projects so reliably disappoint. An organization funds a cleansing exercise, cleanses the data, declares victory, and finds that the data is dirty again within eighteen months, because nothing was done about the processes that were producing dirty data in the first place. Cleansing without validation is drainage without plumbing: it removes today's water and does nothing about the leak. The organizations that solve this permanently are the ones that treat the cleansing as the necessary first step and the entry controls as the actual programme, rather than the reverse.

09 Target Canada and the cost of rushed data

The argument to this point has been statistical. It is worth grounding it in the single most instructive case in the modern record, because it demonstrates that data quality is not a hygiene issue but a solvency issue, and that an organization can lose billions of dollars and an entire market to a problem that would have been visible in a spreadsheet.

When the retailer Target expanded into Canada, it stood up a new instance of a major enterprise platform to run a supply chain across more than a hundred new stores. The platform was proven. The company was sophisticated, ran a large and successful operation in its home market, and understood retail supply chains as well as anyone. What it did not have was time, and so tens of thousands of product records, the dimensions, the weights, the pack sizes, the codes, the prices, were entered under enormous schedule pressure by people working against a deadline. Reporting on the failure subsequently put the accuracy of that data at an estimated thirty percent, against the ninety-eight to ninety-nine percent accuracy the company achieved in its home operations. Figure 9 shows the gap.

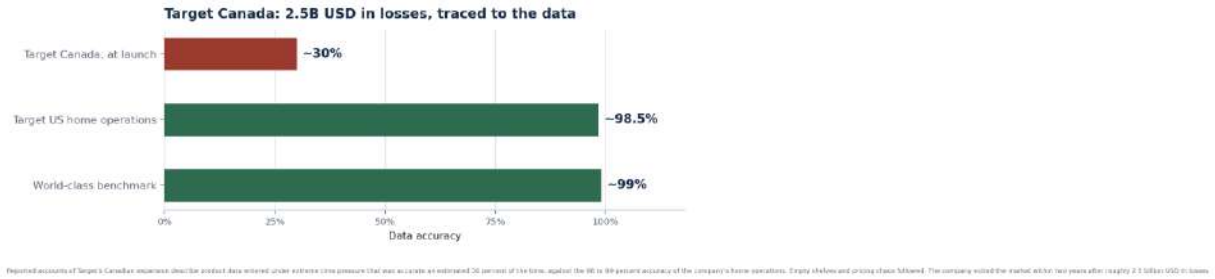

Figure 9. The data accuracy behind one of the most expensive retail failures on record. The systems worked. The data did not.

What followed was not a technology failure in any sense that a technologist would recognize. The systems did what they were told. Replenishment ordered against the numbers it had, which meant that products the system believed were selling were not replenished and products it believed were not selling were. Distribution centers could not process what they could not identify. Shelves were empty in stores while inventory sat in warehouses. Prices were wrong. The supply chain, which is to say the entire operating model of a retailer, could not function, because the data on which every automated decision depended was wrong about one item in three. Less than two years after entering the market the company withdrew, having accumulated approximately two and a half billion dollars in losses, and closed its entire Canadian operation.

Three lessons deserve extraction, and they generalize far beyond retail. First, the data was the project, and it was treated as a task. Nobody funded product data as a workstream with an owner and a quality bar; it was assumed to be data entry, and data entry is what it received. Second, the pressure that produced the bad data was schedule pressure, which is the most common cause of data failure precisely because the data work is the easiest thing to compress when a deadline approaches and it is the last thing whose compression is visible. And third, the failure was invisible until it was catastrophic, because nobody was measuring accuracy against physical reality until the shelves were empty, at which point the measurement was being taken by customers.

A fourth lesson deserves separate treatment because it is the one most applicable to organizations that will never open a store in another country. The failure was a data migration failure, and data migration is the single most under-scoped workstream in enterprise software, present in every implementation and funded in almost none of them. Research on enterprise programs consistently finds that around half of organizations underfund data migration, and the reason is structural: migration looks like a technical task of moving records from one place to another, which sounds like a job for a script. It is not that. It is the work of establishing what the records should say, which requires knowing what they mean, which requires the master data decisions the organization has been deferring for a decade. Migration is where those deferred decisions come due, all at once, on a schedule, and organizations that have not made them will make them badly and under pressure.

The pattern generalizes into a warning that applies to every reader of this article. The moment of maximum data risk in any organization is a system implementation, because it is the moment at which data is created, transformed, and moved at volume, under schedule pressure, by people who are being measured on go-live rather than on accuracy. Any organization currently running an implementation is, right now, creating the data that its automated systems will depend on for the next decade, and it is almost certainly creating it faster than it is validating it. The intervention is simple and is almost never taken: fund the data workstream as a workstream, with an owner and a quality bar, and give it the authority to delay the go-live if the bar is not met. Organizations that will not grant that authority are, in effect, deciding that the go-live date matters more than the data, which is a decision they are entitled to make and should at least make consciously.

10 The five pathologies

Data quality problems are not a single condition, and treating them as one is why remediation efforts so often address the wrong thing. There are five distinct pathologies, each with a different cause, a different symptom, and a different remedy, and an organization that can name which one it has is most of the way to fixing it.

1. **Duplication.** The same real-world entity exists as multiple records. The same customer, entered three times with three spellings. The same supplier under a trading name and a legal name. The symptom is understated volume per entity and double-counted totals. The remedy is entity resolution at the point of creation, not deduplication after the fact, which is the ten in the one-ten-one-hundred rule.
2. **Inconsistency.** The same attribute is described differently in different systems. Different codes, different units, different hierarchies. The symptom is that two systems produce two different correct answers to the same question. This is the master data pathology, it is the most destructive for artificial intelligence, and the remedy is a canonical definition with a single authoritative source, which is an organizational decision before it is a technical one.
3. **Incompleteness.** The fields the business actually needs are not populated. A product without a unit of measure. A supplier without a lead time. A customer without a segment. The symptom is a model that cannot use the feature that matters most. The remedy is making the field mandatory at entry, which is unpopular and effective.
4. **Staleness.** The record was correct and is no longer. Business contact data decays at roughly thirty percent a year, and the same is true of supplier lead times, carrier rates, and cost standards. The symptom is a system that is confidently describing a world that has moved. The remedy is an explicit refresh cadence with an owner, because nothing about a stale record announces itself.
5. **Inaccuracy.** The record does not match physical or contractual reality. The count is wrong, the weight is wrong, the price is wrong. This is the pathology everyone pictures and it is, for most organizations, not the largest of the five. The remedy is measurement against reality, cycle counting for inventory and audit for contracts, and it is the only pathology whose severity can be established without argument.

The diagnostic value of the taxonomy is that the remedies are not interchangeable. An organization suffering from inconsistency will get nothing from a cycle-counting program, because its records are individually accurate and collectively incoherent. An organization suffering from staleness will get nothing from a deduplication exercise. And an organization that has not asked which pathology it actually has will buy a data quality tool, point it at the estate, generate a large report of anomalies, and discover that the tool has found the typographical errors, which were never the problem, and has said nothing about the semantic divergence, which was.

The pathologies compared

The table below sets the five pathologies against their symptoms and their remedies, because the single most common failure in this discipline is applying the remedy for one to the symptoms of another.

Pathology	Symptom	Remedy	Damage to AI
Duplication	Volume per entity understated	Entity resolution at creation	Severe
Inconsistency	Two systems, two correct answers	Canonical definition, one master	Severe
Incompleteness	The field that matters is empty	Mandatory at entry	Severe
Staleness	Confidently describing a world that moved	Refresh cadence with an owner	Moderate
Inaccuracy	Record does not match the shelf	Measure against physical reality	Moderate

Note the pattern in the final column. The two pathologies that do the most damage to an artificial intelligence program, duplication and inconsistency, are the two that a conventional data quality tool is least equipped to find, because neither produces a record that is detectably wrong. They produce records that are locally correct and globally incoherent, and detecting that requires the organization to have decided what coherence means, which is a governance act rather than a scan.

The pathologies also have different visibility profiles, which is why organizations misjudge their own position so consistently. Inaccuracy is the loudest: an empty shelf is impossible to ignore, so organizations that have an inventory accuracy problem generally know it. Staleness is quieter but eventually announces itself, when a supplier lead time that has been wrong for a year finally causes a stockout severe enough to investigate. But duplication and inconsistency are silent, and can persist for a decade without anyone noticing, because every individual record looks fine and the aggregate distortion is invisible unless somebody goes looking for it. An organization that has never deliberately audited its master data for duplicates and definitional divergence should assume it has both, in quantity, and should be especially wary of any conclusion drawn from data it has not audited.

There is a final lesson from the Canadian failure that applies directly to the artificial intelligence programs now being launched, and it concerns the direction of the causal arrow. It would be easy to read the case as evidence that the company should have bought a better system, or a better data quality tool, or hired a better integrator. It should not have done any of those things. What it needed was to slow down: to accept that the data required to run a hundred stores could not be created accurately in the time allowed, and either to allow more time or to open fewer stores. That option was available and was not taken, because the schedule had been announced. Every organization currently promising an artificial intelligence capability by a date, on a foundation it has not measured, is standing in the same position and has the same option, and will face the same choice about whether the date or the data matters more.

11 Why data quality programs fail

Given how well understood this problem is, and how long it has been understood, the more interesting question is not why data is bad but why the programs intended to fix it fail so reliably. The failure modes are consistent, and none of them is technical.

Nobody owns it

Data quality is everyone's problem, which means it is nobody's job. The business team that enters a record does not experience the consequence of entering it badly, because the consequence lands three systems downstream, in a different function, months later. The technology team that stores the record has no authority over how it was created and no knowledge of what it ought to have been. And so the data degrades in the gap between the people who create it and the people who suffer from it, and no organizational structure connects them. Research on governance initiatives repeatedly finds unclear ownership as the leading cause of failure, and the finding is not surprising: a discipline that requires sustained behavior change across many teams cannot be delivered by a function that has authority over none of them.

It is treated as a project rather than an operating discipline

A cleansing project has a start, an end, and a budget, which makes it comfortable to fund and comfortable to declare complete. But data quality is not a state; it is a condition maintained against constant decay, because the business keeps creating new records, the world keeps changing the ones that exist, and every system change introduces new opportunities for divergence. An organization that runs a cleansing project and then disbands the team has purchased a temporary improvement at permanent cost, and it will be back in the same position within two years, at which point it will conclude that data quality projects do not work.

It is scoped to everything, and therefore delivers nothing

The instinct on discovering the scale of the problem is to fix all of it, and the resulting program is so large, so slow, and so undifferentiated that it delivers no visible value for two years and is cancelled in the first budget cycle that gets tight. Not all data is equally important. A small number of critical data elements, the product identifier, the unit of measure, the inventory position, the supplier lead time, drive most of the decisions and most of the damage. An organization that fixes twenty fields that matter, and can prove the improvement, will be funded to fix the next twenty. An organization that sets out to fix everything will be defunded before it has fixed anything.

The improvement cannot be attributed

This is the deepest problem, and it is why the discipline is chronically starved even in organizations that understand it. When data quality improves, other things get better: forecasts become more accurate, planners spend less time reconciling, fewer orders are wrong. But those improvements are attributed to the planning system, the new process, or the good quarter, because attribution flows to whatever is visible, and the data work never is. The organizations that sustain investment here are the ones that

establish a baseline before they start, so that they can point afterward to a number that moved and say, credibly, that they moved it.

It is worth adding a sixth pathology that is not a property of the data at all but of the organization, because it defeats more programs than any of the five. It is the absence of a shared definition. Ask three functions in a supply chain what on-hand inventory means and receive three answers: does it include stock in transit, does it include stock allocated to an order, does it include quarantined stock, does it include the material at the contract manufacturer. Each function has an answer that is right for its purposes, and none of them is written down, and every system implements one of them silently. Every report that reconciles is reconciling by accident. This is not a data quality problem that a tool can find, because every number in every system is correct. It is a definitional vacuum, and it is filled by whoever wrote the code.

The attribution problem has a practical solution that is worth spelling out, because it is the difference between a program that survives and one that does not. Before any remediation begins, pick one or two business outcomes that the bad data is plausibly damaging, and measure them: forecast accuracy for a specific category, availability at a specific set of stores, the hours per week a specific team spends reconciling. Fix the data behind that narrow domain. Then measure the outcomes again. The resulting before-and-after, on a business metric that the organization already cares about, is the only artifact that reliably converts data quality from a technology request into a business investment, and it is available to any organization willing to spend three months proving the case on a small scope before asking for the budget to do it at large scope.

A last word on scope discipline, because it is the failure that kills the most promising programs. The organization that discovers the scale of its data problem feels an entirely reasonable urge to fix all of it, and that urge is the enemy. A program scoped to the whole estate has no deliverable for two years, no measurable improvement to point to in the first budget review, and no defenders when the budget tightens. A program scoped to twenty fields in one distribution center has a result in a quarter, a number that moved, and a sponsor who will fund the second phase because the first one worked. The discipline is not to think small; it is to sequence, so that the credibility earned by each phase pays for the next. Every successful data quality program the authors are aware of began smaller than the problem and grew because it worked.

The remedies also differ in how quickly they pay back, which matters for sequencing. Fixing incompleteness is fast: make the field mandatory, and the data improves from the next record onward, though the historical gap remains. Fixing inaccuracy is moderate: begin cycle counting, and accuracy climbs steadily over a quarter or two. Fixing staleness is fast once an owner exists and slow until one does. But fixing inconsistency is slow under any circumstances, because it requires the organization to make decisions it has spent years avoiding, and those decisions have losers. The practical consequence is that a program should start the inconsistency work first, because it takes longest, while banking the quick wins from the other pathologies to fund the patience the long one requires.

12 Measuring what is actually broken

It follows from the previous section that measurement is not a supporting activity in a data quality program. It is the program's foundation, its funding case, and its only defense against being cancelled. And it is where most organizations begin badly, by attempting to measure everything and therefore measuring nothing that anybody acts on.

The discipline that works starts by identifying the critical data elements: the small set of fields on which the organization's most consequential automated decisions actually depend. In a supply chain this list is short and largely predictable. The item identifier and its master attributes. The unit of measure and pack configuration. The on-hand inventory position by location. The supplier lead time. The customer ship-to and its associated service terms. The cost standard. Somewhere between twenty and fifty fields, in most organizations, carry the overwhelming majority of the decision weight, and the remaining thousands, while not worthless, do not merit the same rigor.

For each critical element, the organization then establishes a measurement against ground truth, and the phrase against ground truth is the entire point. Inventory accuracy is measured by counting the shelf, not by comparing one system to another, because two systems can agree and both be wrong. Supplier lead time accuracy is measured against what actually arrived. Product weight is measured on a scale. This is laborious, and it is the only kind of measurement that cannot be argued with, which is precisely why it is the kind that gets budgets approved. A report showing that two systems disagree provokes a debate about which one is right. A report showing that the system says four hundred units and the shelf holds two hundred sixty ends the debate.

The measurement should then be published, on a cadence, with an owner named against each element, and this is where most of the organizational work actually happens. Publishing a number attaches accountability to it. The moment the head of a function sees, in a report that their peers also see, that the data their team creates is accurate seventy percent of the time, the incentive structure changes, and it changes far more effectively than any amount of exhortation about the importance of data. The measurement is not the diagnosis. It is the intervention.

A note on the tooling, because the question always arises and the honest answer is unfashionable. Data quality software is useful and is not the answer. Profiling tools that scan an estate and report anomalies are truly valuable for the first pass, because they will find the empty fields, the impossible values, the format violations, and the obvious duplicates faster than any human. Master data platforms provide real machinery for maintaining golden records once the organization has decided what a golden record is. Observability tools detect pipeline failures that would otherwise pass unnoticed. All of this is worth buying. None of it decides which system is authoritative for a product, none of it defines what on-hand inventory means, and none of it makes anybody accountable. The tools automate the enforcement of decisions the organization has made. They do not make the decisions, and an organization that buys a tool in the hope that it will is buying a report it will not act on.

A final measurement point concerns drift, and it is the one that separates a program from a project. Data quality does not stay fixed, because the business does not stay still: new products are launched,

new suppliers are onboarded, new systems are connected, and each is an opportunity for the definitions to diverge again. An organization that measures its critical elements once has taken a photograph. An organization that measures them on a cadence has installed an instrument, and the instrument will show, within a quarter or two, whether the entry controls are actually holding or whether the improvement is already eroding. Measuring once tells you where you are. Measuring repeatedly tells you whether you are winning, and only the second question has an answer that anyone can act on.

Two organizational units are worth building and are often skipped. The first is a small data quality function, of perhaps two or three people, whose entire job is measurement, publication, and the maintenance of the critical element register. This is not a large investment and it is the difference between a program and a memo. The second is a set of data stewards embedded in the business functions that create the data, part-time and named, who are accountable for the quality of what their function produces and who have the standing to change how their colleagues work. Neither of these is expensive. Both are the machinery through which every other recommendation in this article is actually executed, and an organization that adopts the recommendations without the machinery will find that nothing happens.

13 A remediation sequence that works

For an organization that has decided to act, the order of the work matters as much as the work itself, and the natural order, the one an organization arrives at by instinct, is close to the reverse of the right one. What follows is a sequence that has the useful property of producing a visible return early enough to survive the second budget cycle.

1. **Pick the critical data elements.** Name the twenty to fifty fields that drive the consequential automated decisions. Resist the pull toward comprehensiveness. Scope discipline at this step is what determines whether the program is alive in two years.
2. **Measure them against ground truth, and publish the number.** Establish the baseline before anything is fixed, because it is both the diagnosis and the evidence that will later prove the program worked. A number nobody has published is a number nobody will fund.
3. **Fix the entry controls before cleansing anything.** Validation at the point of creation is the one in the one-ten-one-hundred rule, and cleansing without it is drainage without plumbing. Make the critical fields mandatory, add the duplicate check at creation, enforce the code list. This step is unpopular and is the highest-return action available.
4. **Establish the authoritative source for each master entity.** Decide, as an organizational matter, which system is the master for the product, the customer, the supplier, and the location, and require every other system to defer to it. This is a governance decision, not a technology purchase, and it is the one that fixes inconsistency, which is the pathology that most damages artificial intelligence.
5. **Cleanse, once, against the now-defended perimeter.** With entry controls in place and an authoritative source defined, remediation of the historical stock is finally worth doing, because it will not immediately re-dirty. Doing this step first, which is what most organizations do, is what makes data quality programs look futile.
6. **Assign an owner to each critical element and keep publishing.** Data quality is maintained, not achieved. An owner, a measurement, and a published cadence is the machine that maintains it, and the absence of that machine is why the previous cleansing project did not last.
7. **Only then, automate.** The advanced planning system, the optimization engine, the artificial intelligence agent: these become worth deploying once the data beneath them can bear their weight. Deployed before, they will fail, and the failure will be attributed to the vendor rather than to the foundation.

The sequence is not complicated, and none of its steps require technology the organization does not already own. What it requires is the willingness to spend the first year of an artificial intelligence budget on something that is not artificial intelligence, which is a political act rather than a technical one, and which is precisely why so few organizations manage it.

The quickest way to make this real for an executive audience is a demonstration rather than a presentation, and it costs a morning. Take the ten highest-value items in a single distribution center. Print what the system says is on hand. Walk to the racks and count. Bring both numbers to the next leadership meeting. In the great majority of organizations the two columns will not match, and the gap

between them will do more to unlock a data budget than any industry statistic ever assembled, because it is not a claim about the industry. It is a claim about this company, verifiable by anyone in the room, and it cannot be dismissed as vendor marketing or analyst hype. The most persuasive argument in this entire field is a clipboard.

Two mistakes are worth flagging in the execution of this sequence, because they are the ones that most commonly derail it. The first is skipping step three, the entry controls, because they require asking business users to do more work and the political cost of that request is immediate while the benefit is deferred. An organization that skips it will complete a cleansing project and watch the data degrade again, and will conclude, wrongly, that the discipline does not work. The second is attempting step four, the authoritative source decision, as a technical exercise. It is not one. Deciding which system is the master for the product record is a decision about which function owns the product, and it will be resisted by whichever function loses, and it requires an executive with the standing to make the call. Delegating it to an architecture committee guarantees that it will be discussed indefinitely and never decided.

The sequence also implies a budget shape that is worth making explicit, because it differs sharply from how these programs are usually funded. The conventional shape puts almost all of the money into the technology, with a small allowance for data migration treated as a task. The shape this article recommends inverts that for the first year: most of the money goes into measurement, entry controls, master data decisions, and remediation, with the automation deferred until the foundation can bear it. That is a harder budget to sell, because it delays the visible thing, and it is the budget that actually delivers the visible thing, because the alternative is to buy the capability and then discover that it cannot be used. A leader who can hold that line for one budget cycle will be rewarded in the second.

14 Governance, ownership, and a scoring rubric

The practices above will not survive without a governance structure, and the structure that works is smaller and less bureaucratic than the phrase implies. It consists of three things: a named owner for each critical data element, a published measurement against ground truth, and a forum with the authority to require a function to fix the data it creates. Nothing else in the apparatus of data governance matters as much as those three, and organizations that build elaborate councils without them produce documentation rather than data.

What to demand of a vendor

The data question also belongs in every software procurement, and it is almost never asked. Before buying any system that will consume or produce master data, a buyer should establish in writing what the system requires as input, what quality level it assumes, and what happens when that level is not met. A planning vendor that cannot state the inventory accuracy its optimization assumes is selling a capability it has not thought about. A vendor whose artificial intelligence features require clean master data, and most do, should be asked directly what the model does when the data is inconsistent, and the honest answer, which the better vendors will give, is that it produces a confident wrong answer. That admission is not a reason to walk away. It is the beginning of an accurate business case, in which the data work is scoped and funded rather than assumed.

A scoring rubric

The dimensions below allow an organization to score its own readiness before it commits to an automation program, and to score a vendor's honesty about what its system requires.

Dimension	What good looks like	Red flag
Critical elements defined	A named list of the fields that drive decisions	An intention to improve data quality generally
Measured against reality	Inventory counted; lead times verified against receipts	Systems compared only to other systems
Entry validation	Mandatory fields, duplicate checks, enforced code lists	Cleansing projects with no entry controls
Authoritative source	One system is master for each entity, by decision	Every system holds its own version
Named ownership	A person accountable for each critical element	Data quality owned by a committee
Published cadence	Quality levels reported where peers can see them	Measured privately, or not at all
Vendor candor	States the data quality its system assumes	Claims the system handles messy data

The last row is the one that catches the most vendors and deserves emphasis. A vendor who claims that its system handles messy data is either using a phrase that means nothing or is making a claim that is

false, because no system infers a fact that its inputs do not contain. The better vendors will tell a buyer plainly what quality level their capability assumes and will help scope the work required to reach it, and that candor is worth more in a partner than any feature on the comparison sheet.

A word on where the ownership should sit, because organizations get this wrong in a predictable way. The instinct is to give data quality to the technology function, on the reasoning that data lives in systems and systems belong to technology. This fails, reliably, for a reason that is obvious in retrospect: the technology function does not create the data, does not know what it should say, and has no authority over the people who do. Data is created by the business, in the act of receiving a shipment, onboarding a supplier, launching a product, or entering an order, and it can only be improved by the business changing how it performs those acts. The right structure gives ownership of each critical data element to the business function that creates it, gives the technology function responsibility for the controls and the measurement, and gives somebody senior enough to compel both the authority to hold them to it.

The final governance point concerns the artificial intelligence program specifically, because it is where the demand for data quality will now originate whether the organization plans for it or not. Every serious automation initiative will discover, somewhere between the pilot and production, that its data is not adequate, and at that point one of two things happens. Either the initiative quietly narrows its ambitions until it is doing something the data can support, which is how most pilots end, or it becomes the forcing function that finally funds the foundation, which is how the good ones do. Which of these occurs is not determined by the technology or the vendor. It is determined by whether anyone in the room is willing to say that the honest answer to why the pilot failed is that the organization was not ready, and to ask for the year it would take to become so.

One further vendor question is worth adding, because it separates the serious partners from the rest with a single sentence. Ask what the system does when it encounters data it cannot reconcile. A weak answer describes error logs. A strong answer describes exception handling: the system detects the irreconcilable case, refuses to act on it, routes it to a human, and does not silently guess. The distinction matters enormously in an agentic context, because a system that guesses when it is uncertain will produce confident wrong actions at machine speed, whereas a system that knows the limits of its own inputs will produce a manageable queue of exceptions. That property is not a feature on any comparison sheet, and it is worth more than most of the features that are.

It is worth acknowledging the objection that this article invites, because a fair-minded reader will raise it. If the data has been this bad for this long, and organizations have nonetheless functioned, perhaps the problem is being overstated. The answer is that they functioned because of the human error-correction layer described at the outset, and that they paid for it in a currency nobody counted: the time of skilled people reconciling, checking, and overriding. A majority of organizations report losing a day or more each week to master data problems. That is not a system working; that is a system being carried. And the reason the situation is now urgent is that the organizations removing the carriers, in favor of automation, are the same organizations that never counted what the carrying cost, and are therefore unable to see what they are about to lose.

A short checklist is worth stating for the reader who will act on this article on Monday. Commission a physical count of the top items in one facility and compare it to the system. Ask, of the three most important automated decisions the organization makes, which fields they depend on and who owns each. Ask the last vendor that demonstrated an artificial intelligence capability what data quality its system assumes, and note whether they can answer. Find out whether anyone measures inventory accuracy at all, and if so, whether the number is published anywhere that the people who create the data can see it. Four questions, none of which requires a consultant, and the answers will tell a leader more about the organization's readiness for automation than any assessment they could commission.

It is worth being explicit that none of this is an argument for delay in the sense that vendors will characterize it. An organization that adopts these practices is not standing still while its competitors advance; it is doing the work that determines which of them will get value from the technology and which will merely have purchased it. The competitive advantage in this cycle will not accrue to the company that deployed the most models. It will accrue to the company whose models were fed data that described reality, and that company will pull ahead precisely at the moment when the others discover that their pilots do not scale. The foundation is not the thing that delays the advantage. It is the advantage.

The last observation is about culture, and it is the one that determines whether any of this survives. In most organizations, the person who reports that the data is bad is treated as a problem, because they have introduced friction into a schedule that everyone is being measured against. In the organizations that solve this, that person is treated as an asset, because the alternative to hearing the bad news early is discovering it late, at scale, in production, in front of a customer. Changing which of those two responses is the default is not a governance intervention or a technology one. It is a leadership one, it is free, and it is probably the single highest-return decision available to anyone reading this article.

15 Conclusion: fund the foundation

The reckoning described in this article is not a prediction. It is already happening, in the form of artificial intelligence pilots that produce no value, planning systems that were supposed to transform performance and did not, and agents that make confident decisions on inputs that nobody has verified. The organizations experiencing these failures overwhelmingly attribute them to the technology, and in most cases they are wrong. The technology arrived working. It was deployed onto a foundation of records that were accurate two-thirds of the time, inconsistent across systems, missing the fields it needed, and owned by nobody, and it did exactly what such a system will always do, which is to compute a precise answer to a question the data could not support.

The correction is not glamorous and it is not fast. It consists of naming the twenty fields that matter, measuring them against physical reality, publishing the result where the people who create the data can see it, putting validation at the point of entry, deciding which system is authoritative for each master entity, and giving every critical element an owner with a name. That is the whole program. It requires no new technology, it can begin next week, and it will produce, within a year, a measurable improvement in forecast accuracy and inventory availability that the organization can bank before it spends anything at all on artificial intelligence.

The final argument is one of sequence, and it is the one worth taking to the board. The data foundation is not a prerequisite that delays the artificial intelligence program. It is the artificial intelligence program, or at least the first year of it, because there is no version of an automated supply chain that works on data that is right two-thirds of the time, and every dollar spent on a model that will consume such data is a dollar spent on a confident error. The organizations that will get value from this technology are not the ones that adopted it first. They are the ones that could feed it. That is a less exciting sentence than the vendors are offering, and it is the one the evidence supports.

It is worth ending on the note that this article is, despite its severity, an optimistic one. The problem it describes is entirely solvable, requires no technology that does not already exist, and the interventions that solve it are among the cheapest available to any technology organization. A validation rule at the point of entry costs a developer a day. A physical count of a warehouse costs a morning. A decision about which system is master for the product record costs an executive a difficult conversation. None of this is a research problem or a capital project. What it requires is that somebody with standing decide that the foundation matters more than the next feature, and then hold that position through the eighteen months in which the work is invisible. That is a leadership problem, and leadership problems, unlike technical ones, can be solved by deciding to solve them.

The question to carry from this article into the next steering committee is one sentence long and will not be comfortable to ask. What is our actual inventory accuracy, measured against a physical count taken this quarter, and does anyone in this room know the number. In most organizations nobody will know, which is itself the finding, and the silence that follows the question is the beginning of the program. The organizations that will succeed with automation over the next five years are not the ones that moved

earliest or spent most. They are the ones that could answer that question, and that did something about the answer.

16 Methodology, caveats, and sources

Methodology

- This article synthesizes analyst research, academic studies, industry benchmarking, and documented failure cases, current to mid-2026. Supply Chain Research is independent and accepts no payment from the vendors, consultancies, or platforms discussed.
- Particular attention has been paid to the provenance of widely repeated statistics in this field, which is unusually prone to large, round, unattributable figures. Where a number is a self-reported estimate rather than a measurement, this is stated.

Caveats

- The frequently quoted figure of twelve point nine million dollars as the annual cost of poor data quality originates in a 2020 analyst vendor-landscape study and represents the average self-estimate of roughly one hundred fifty reference customers of data quality vendors. It is a self-selected sample of organizations already purchasing data quality software. It is a reasonable order of magnitude for a large enterprise and is not an audited measurement.
- Inventory and forecast accuracy figures come from different sources measuring different populations with different methods, and are not directly comparable to one another. They are presented to convey the gap between actual and required accuracy, not as a single benchmark.
- The one-ten-one-hundred rule is a practitioner heuristic, not a measured finding. The multiples are directional and are used here to convey the shape of the cost curve rather than to price a specific remediation.
- The estimate that master data programs frequently fail to meet objectives is attributed to analyst commentary reported at second hand and should be treated as directional. The Target Canada figures are drawn from press accounts and subsequent case analyses rather than from company disclosure of data accuracy specifically.

Sources

[1] Gartner. Data quality: why it matters and how to achieve it (origin of the widely cited annual cost figure).

[2] Gartner. Why generative AI projects fail, including poor data quality among the leading causes.

[3] Auburn University RFID Lab. Research on retail inventory accuracy.

[4] CAPS Research. Cross-industry benchmarking of inventory accuracy.

[5] McKinsey. Research on master data management maturity and enterprise data foundations.

[6] Henrico Dolfing. Project failure case study: Target Canada.

[7] MIT Sloan Management Review. Research on the revenue impact of poor data quality.

[8] Supply Chain Management Review. Retail has an inventory accuracy problem.

Additional context drawn from Fluent Commerce survey research on retailer inventory accuracy; from industry reporting on demand forecast accuracy by sector; from practitioner literature on the one-ten-one-hundred rule in data quality management; and from analyst commentary on master data management program outcomes. Self-reported and vendor-sourced figures are identified as such and are directional. This article is analysis, not legal, procurement, or investment advice.

Supply Chain Research is an independent, vendor-neutral research platform for supply chain and technology leaders. We accept no payment from the vendors, consultancies, or platforms discussed. This article is analysis, not legal, procurement, or investment advice, and its conclusions should be validated against your own requirements before any decision.