

SUPPLY CHAIN RESEARCH · EDITORIAL FEATURE

The Master Data Mandate

Who Actually Owns the Definition of a Customer

Ask five departments what a customer is and get five answers. Master data management is sold as a technical project, buy the software, build the golden record, but it is a political settlement over whose definition wins. That is why it fails so often, and why it now blocks AI.

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

Supply Chain Research | supplychainresearch.com | 2026

Contents

Who Actually Owns the Definition of a Customer

01	Executive summary	01
02	What is a customer, exactly?	02
03	The failure statistics, and why they disagree	03
04	A political settlement, not a technical project	04
05	The authority gap and the vanishing owner	05
06	Centralized, federated, and the golden-record trap	06
07	Data mesh and governed multiplicity	07
08	Why this is now an AI problem.....	08
09	The identity spine and the governed view	09
10	The vendor claims, and what software cannot do.....	10
11	The fairness case: some multiplicity is real	11
12	Settle the definition, do not legislate it away	12
13	A governance protocol, and a scoring rubric	13
14	Minimum-viable governance, not the enterprise ocean	14
15	Conclusion: name the owner, keep the views	15
16	Methodology, caveats, and sources.....	16

01 Executive summary

Ask the sales, finance, marketing, support, and legal functions of a large enterprise to define a customer, and receive five different, mutually incompatible, and individually correct answers. Sales means the account or opportunity, finance means the paying legal entity, marketing means the individual contact, support means the user or seat, and legal means the corporate hierarchy. Each definition is right for its purpose, and each is embedded in that function's systems, reports, and processes. The trouble is that every cross-functional report, every enterprise metric, and increasingly every artificial-intelligence system assumes a single shared definition of a customer, and none exists. The numbers do not reconcile, the analyses rest on a fiction, and the models are trained on an ambiguity they assume away.

This article argues that master data management, the discipline meant to resolve this by establishing agreed definitions of the core entities an enterprise shares, fails so frequently not because the software is inadequate but because the problem is misdiagnosed. Master data management is sold and bought as a technical project, buy the platform, build the single golden record, when it is fundamentally a political settlement over whose definition of a shared entity prevails and who holds authority to decide. Software cannot adjudicate a turf dispute between functions, which is why, as one prominent analyst has put it, implementing master data software does not fill the master data gap. We say honestly that some of the multiplicity is legitimate and that a forced single definition can destroy real information. But the settlement must still be made, explicitly and politically, and this article sets out how: not by legislating away the differences with a golden record that forces false consensus, but by separating a shared canonical identity from the governed, function-specific views that legitimately differ, and by naming an accountable owner with the authority and the tenure to make the settlement hold, which is precisely what most organizations fail to do.

75-90%

the range of quoted MDM failure rates across sources, none able to measure it consistently

~2.5 yrs

average chief data officer tenure, against roughly 7 for the CEO, on a problem that takes years

62%

of organizations citing data governance as the biggest barrier to deploying AI (Gartner)

The argument in brief

- **There is no single definition of a customer.** Five functions define the entity five incompatible ways, each correct for its purpose. Reporting and AI assume one definition; the absence of one is why numbers do not reconcile.
- **MDM is a political settlement, not a technical project.** It decides whose definition wins and who holds authority. Software cannot adjudicate a turf dispute, which is why buying an MDM platform does not close the gap.
- **The accountable owner vanishes too soon.** The chief data officer, carved from the CIO's remit and now threatened by the chief AI officer, serves about 2.5 years, and the political settlement takes longer than the owner lasts.

- **The golden record forces false consensus.** A single centralized record tries to legislate away differences that are legitimate. Federated governance, a shared identity spine with governed local views, preserves the valid nuance while making it reconcilable.
- **Settle it, do not automate it away.** Name the political owner, separate canonical identity from function-specific views, enumerate the legitimate views, govern definitions as products, and start with the few entities that matter.

02 What is a customer, exactly?

The entire problem this article addresses is contained in a question that sounds trivial and is not: what, exactly, is a customer? The question seems to have an obvious answer until one asks it of the different functions of an enterprise, at which point the obvious answer fragments into several incompatible ones, each held with conviction by the function that holds it. Figure 1 sets out the fragmentation.



See that functions often a customer is and get the different answers. Sales means the account, finance means the paying legal entity, marketing means the individual contact, support means the user or seat, and legal means the corporate hierarchy. Each is correct for its purpose. The problem is that reporting, analytics, and any AI system that needs functional design areas (contacts, and even more, across business, for the inclusion and the model) will be a problem.

Figure 1. One word, five incompatible answers. Sales means the account, finance the paying entity, marketing the individual, support the seat, legal the hierarchy. Each is correct for its purpose; reporting and AI assume a single definition that does not exist.

Consider each function's answer and why it is correct for that function. Sales defines a customer as the account or the opportunity, because sales works in terms of deals with organizations and manages a pipeline of accounts, and for sales the customer is the entity it is selling to, which is the account. Finance defines a customer as the paying legal entity, because finance must invoice, collect, and recognize revenue against a legal party that can be billed and held to a contract, and for finance the customer is whoever pays, which is a specific legal entity that may differ from the account sales sold to. Marketing defines a customer as the individual contact, because marketing communicates with people, tracks individual engagement, and nurtures individual relationships, and for marketing the customer is the person, not the organization. Support defines a customer as the user or the seat, because support serves individuals using the product and tracks entitlements at the level of the user or the licensed seat. And legal defines a customer as the corporate hierarchy, because legal must understand the full structure of related entities for questions of liability, contracting, and compliance.

None of these definitions is wrong. Each is precisely correct for the purpose its function serves, and each is embedded, correctly, in that function's systems and processes. The sales system is built around accounts because sales works with accounts; the finance system is built around payers because finance bills payers; and so on. If each function existed in isolation, the multiplicity of definitions would cause no problem, because each would use its own correct definition for its own correct purpose, and the definitions would never need to reconcile. The problem arises entirely from the need to combine information across functions, which requires the definitions to relate to one another, and they do not relate cleanly because they carve the world differently.

The consequences appear the moment anyone tries to answer a question that spans functions, which is most questions that matter. How many customers does the enterprise have? The answer depends on whether a customer is an account, a payer, a contact, a seat, or a hierarchy, and the five functions will give five different counts, each correct by its own definition and none reconcilable with the others. What is a customer worth? The lifetime-value calculation must decide whether it is aggregating accounts, payers, or contacts, and blending them produces a number that means nothing. Which customers are at risk of churning? The churn model must know what a customer is to count one as lost,

and if it uses one function's definition while another function still considers the entity active, the model miscounts. Every cross-functional question about customers founders on the absence of a shared definition, and the enterprise discovers that a word it uses constantly, and assumed it agreed on, means five different things, none of which it can reconcile without a settlement it has never made.

The trouble deepens when one notices that the five definitions do not merely differ but overlap partially and inconsistently, so they cannot be cleanly separated either. A single individual might be, simultaneously, a contact in marketing's sense, a user in support's sense, and a signatory of the contract that finance bills, so the same person appears across three functional definitions in three different roles, and the enterprise cannot say whether that person is one customer or three without deciding which definition governs. The corporate hierarchy that legal tracks contains the paying entity that finance bills, which contains the account that sales sold, which contains the contacts that marketing engages, so the definitions nest and overlap in ways that resist any simple mapping. The relationships between the definitions are themselves complex and contested, which means that even reconciling them, let alone unifying them, requires untangling a web of partial overlaps that no single function sees whole.

This partial overlap is why the master data problem cannot be solved by simply picking one definition and translating the others into it, which is a tempting shortcut that does not work. If the definitions were cleanly separable, one could designate a primary definition and derive the others from it by fixed rules, and the problem would be tractable. But because the definitions overlap partially and inconsistently, no such clean derivation exists: one cannot mechanically convert marketing's contacts into finance's payers, because the relationship between contacts and payers is many-to-many and context-dependent, with some contacts associated with multiple payers and some payers associated with multiple contacts. The absence of a clean mapping between the definitions is what makes the problem truly hard, and it is why the settlement this article recommends focuses on establishing a shared identity that all the definitions can reference, rather than on translating the definitions into one another, which the partial overlaps make impossible.

03 The failure statistics, and why they disagree

Master data management, the discipline meant to resolve this, has a reputation for failure, and the statistics quantifying that failure are worth examining, both for what they say and for the more interesting fact of their disagreement. Figure 2 collects several.

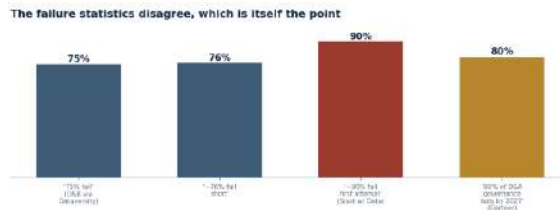


Figure 2. Quoted MDM failure rates of 75, 76, and 90 percent, and a Gartner projection that most data-governance initiatives will fail by 2027. Several figures trace to data-services vendors; that the field cannot measure its own failure rate consistently is the more reliable signal.

The figures circulate widely. Master data programs are said to fail at a rate of 75 percent, a figure attributed to Gartner and propagated through a data-services provider, with the added claim that the rate has worsened in recent years. Others put the figure at around 76 percent falling short of objectives, or state that only about a quarter succeed. Still others claim that roughly 90 percent of master data projects fail on their first attempt. And Gartner has projected that 80 percent of data-and-analytics governance initiatives will fail by 2027, with a related finding that a large majority of organizations cite governance as a barrier to their goals. The figures agree that failure is common and disagree on almost everything else, including how common.

Two cautions are warranted before relying on any of these numbers. First, several trace to parties with a commercial interest in the perception that master data management is hard and failure-prone, namely the vendors of data-management software and services who benefit when enterprises believe they need help, and such figures should be treated as directional rather than precise, in keeping with this publication's practice of flagging interested statistics. Second, and more fundamentally, the definitions of failure vary across the sources, so the figures are not measuring the same thing: one source's failure is another's partial success, and the wide range from 75 to 90 percent reflects inconsistent definitions as much as inconsistent outcomes. The precise numbers should carry little weight.

But the disagreement is itself the most instructive finding, and it should be read as evidence rather than as noise. A field that cannot measure its own failure rate within a range of fifteen percentage points, and whose failure figures come largely from parties that profit from the appearance of difficulty, is a field whose failure is real but whose measurement is unreliable, and the unreliability of the measurement is a clue to the nature of the failure. Master data programs fail in ways that resist clean measurement because their success is not a technical binary, software deployed or not, but a political and organizational condition, definitions agreed and authority established, that is inherently harder to assess and that different observers assess differently. The distinctive difficulty of measuring master data failure points to its nature: it is not a technical outcome that can be cleanly scored but an organizational settlement whose success is a matter of degree and of judgment, which is exactly why the statistics

disagree and exactly why the technical framing that produces clean success metrics for software projects fails to capture it.

The propagation of these statistics repays a moment of attention, because it illustrates a broader pattern in how difficulty-of-a-problem figures circulate in enterprise technology. A figure originates somewhere, frequently in a vendor's marketing or a consultancy's report, attributed to an authoritative source such as a major analyst firm, and it then propagates through repetition, each repeater citing the prior one rather than the original, until the figure is ubiquitous and its provenance is lost. By the time an enterprise encounters the claim that master data programs fail at some high rate, the figure has been repeated so widely that it feels like established fact, even though it may trace back to a single interested source using a particular definition of failure that the repeaters do not examine. The ubiquity of a figure is not evidence of its reliability; it may reflect only the effectiveness of its propagation, and a figure that everyone cites may be one that no one has verified.

This matters for how an enterprise should use such statistics in its own planning, which is carefully and skeptically. The high failure rates are frequently deployed to motivate the purchase of the same software whose category has the high failure rate, an odd rhetorical move in which the difficulty of the problem is used to sell the solution that reportedly fails most of the time, and an enterprise should notice the circularity. The reliable inference from the failure statistics is not that a particular vendor's software will overcome the failure rate, which the statistics do not support, but that the problem is truly hard and commonly mishandled, which argues for understanding why it is hard, the political nature this article describes, rather than for buying the software that the failure-rate marketing promotes. The statistics, read correctly, are an argument for diagnosing the problem properly, not for purchasing the solution that shares the category's failure rate, and an enterprise that lets the failure statistics stampede it into a software purchase has drawn exactly the wrong lesson from them.

04 A political settlement, not a technical project

The central thesis of this article is that master data management fails so frequently because it is misdiagnosed: treated as a technical project when it is fundamentally a political settlement. Understanding this distinction is the key to understanding both the failures and the remedy. Figure 3 states it.

The category error at the root of most MDM failure



Organizational master data management is a technical project, but the software build the single golden record. It is actually a political settlement: deciding whose definition of a shared entity prevails and who fairly authority over it. Software cannot adjudicate a dispute between functions over shared aspects of the truth sets, which is why, as Gartner has noted, implementing MDM software does not fill the MDM gap. The gap is organizational.

Figure 3. The category error at the root of most MDM failure. Organizations treat it as a technical project, buy software and build a golden record, when it is a political settlement over whose definition wins. Software cannot adjudicate a turf dispute.

The technical framing goes like this. The enterprise has inconsistent data about its core entities, so it should acquire a master data management platform, use it to consolidate the scattered records into a single authoritative version, the golden record, and thereby establish one consistent definition that all functions draw from. In this framing, the problem is technical, scattered and inconsistent records, and the solution is technical, software that consolidates them, and the project is a matter of selecting, deploying, and configuring the platform. This is how most master data initiatives are conceived, budgeted, and staffed, as technology projects owned by technology functions and measured by technology milestones.

The framing is wrong because it mistakes the symptom for the disease. The inconsistent records are the symptom; the disease is that the functions do not agree on what the entity is, and their disagreement is embedded in records that reflect their different definitions. Consolidating the records into a single golden record does not resolve the disagreement; it suppresses it, by imposing one definition and discarding the others, and the functions whose definitions were discarded do not stop needing them, because their definitions were correct for their purposes. The golden record, built to end the inconsistency, either fails to be adopted, because the functions continue using their own definitions that the golden record does not serve, or is adopted and breaks the functions whose legitimate needs it does not meet. The technical solution addresses the symptom while leaving the disease, the unresolved disagreement over the definition, untouched.

The disease is political because resolving it requires deciding whose definition prevails, which is a question of authority, not technology. When two functions define a shared entity differently, and the enterprise needs a shared definition for cross-functional purposes, someone must decide which definition governs the shared uses, or how the definitions relate, and that decision advantages some functions and constrains others, which makes it a political act, an allocation of authority over a shared resource. Software cannot make this decision, because it is not a technical question with a technical answer; it is a question of whose needs take precedence in the shared space, and that is a matter of organizational authority and negotiation. This is what the analyst meant in observing that implementing master data software does not fill the master data gap: the gap is the unmade political decision about

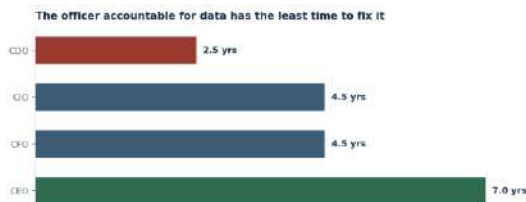
whose definition governs, and software, which can consolidate records but cannot adjudicate authority, cannot fill it. An enterprise that buys software to solve a political problem has bought a tool that cannot address its actual difficulty, which is why the software so often sits deployed while the inconsistency persists, the definitions still unreconciled because the political settlement they require was never made.

The political nature of the settlement has a consequence that organizations find uncomfortable, which is that making it requires someone to lose, at least relative to the status quo in which each function's definition governs its own domain unchallenged. When the enterprise establishes how the definitions relate for shared purposes, it necessarily privileges some functions' framings over others in the shared space, and the functions whose framings are not privileged experience this as a loss of autonomy over how their entities are understood enterprise-wide. This is why the settlement is political rather than merely technical: it redistributes a form of authority, the authority to define shared entities, and redistributions produce winners and losers who contest the outcome. A settlement that pretended no one lost, that claimed to serve every function's definition equally in the shared space, would be evading the actual difficulty, because the shared space cannot simultaneously honor incompatible definitions, and choosing how to relate them advantages some over others.

Recognizing that the settlement produces relative winners and losers clarifies why it requires authority to make. A negotiation in which every party can veto any outcome that disadvantages it will not converge, because every possible settlement disadvantages someone relative to the status quo, so someone will always veto. The settlement therefore requires an authority above the functions that can impose an outcome the functions would not all voluntarily agree to, deciding how the definitions relate in the shared space even over the objection of the functions whose framings are not privileged. This is precisely the authority that the vanishing chief data officer lacks and that the settlement requires: not the authority to run a software project, which the data officer may have, but the authority to impose a redistribution of definitional control that some functions will resist, which requires executive backing above the functional level. Without that authority, the settlement cannot be made, because the functions will not voluntarily converge on any outcome, and the master data effort stalls in an unresolvable negotiation, which is one more reason so many stall.

05 The authority gap and the vanishing owner

If master data management is a political settlement requiring authority to make and sustain, then the question of who holds that authority becomes central, and here the enterprise confronts a structural problem: the executive nominally responsible for data has neither sufficient authority nor sufficient tenure to make the settlement hold. Figure 4 shows the tenure gap.



The average chief data officer serves roughly 2.5 years, against about 4.5 for the CIO and CFO and around 7 for the CEO. The role is frequently carved out of the CIO's remit, creating a turf conflict on day one, and it is now threatened further by the emerging chief artificial intelligence officer, and the one who owns them is gone before the settlement holds. Source: figures from MIT Sloan and industry surveys.

Figure 4. The officer accountable for data has the least time to fix it. The chief data officer serves roughly 2.5 years against about 7 for the CEO, on a problem that takes years to settle politically, and the role is frequently carved from the CIO's remit.

The chief data officer, the role created to own enterprise data, has an average tenure of roughly two and a half years, according to research from MIT Sloan and corroborating industry surveys, against about four and a half years for the chief financial and information officers and around seven for the chief executive. The role turns over faster than any other in the C-suite, and it turns over faster than the master data settlement it is meant to achieve can be made and stabilized, because that settlement is a political process of negotiating definitions and authority across functions that takes years, not months. The owner of the problem is gone before the problem is solved, and the successor inherits a partial settlement, restarts the negotiation, and frequently departs in turn before completing it, so the settlement is perpetually in progress and never stable.

The tenure problem compounds an authority problem baked into the role's origins. The chief data officer position was frequently carved out of the chief information officer's remit, created by separating responsibility for data from responsibility for technology, which means the role begins with a turf conflict, sharing a boundary with a more established officer who previously owned what the new role now claims. This origin leaves the chief data officer with contested authority, responsible for data but dependent on the information officer for the systems that hold it and on the business functions for the definitions that govern it, holding accountability without commensurate authority. And the emerging chief AI officer role now threatens the data officer's remit further, claiming the artificial-intelligence agenda that depends on the data the data officer owns, adding another boundary dispute to a role already short on clear authority.

The combination is debilitating for the master data mission. The settlement requires authority, to decide whose definition governs, and tenure, to see the multi-year negotiation through, and the officer nominally responsible has a deficit of both: contested authority from a role carved out of and bordered by others, and short tenure that ends before the settlement stabilizes. This is a structural reason master data programs fail that has nothing to do with software and everything to do with organizational design: the enterprise has assigned a political mission requiring sustained authority to a role configured with contested authority and short tenure, and then wonders why the mission does not succeed. The

remedy, developed in the constructive sections, is not a better data officer but a different allocation of authority, one that gives the master data settlement an owner with real, uncontested authority and the executive backing to sustain it beyond any individual's tenure, because a political settlement cannot be made by an officer who lacks the authority to decide and lacks the tenure to see the decision through. The vanishing owner with contested authority is a recipe for the perpetual, never-stable settlement that master data programs so often become.

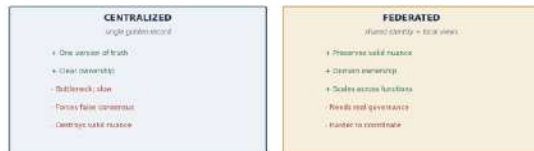
The authority gap has a subtler dimension worth drawing out, concerning the difference between formal and effective authority. A chief data officer may have formal authority over data, granted by the organizational chart, while lacking the effective authority to make functions change how they define their entities, because effective authority over a function's definitions rests with that function's leadership, who control the systems and processes that embody the definitions. The data officer can decree a standard, but if the functional leaders do not choose to adopt it, the decree is inert, because the data officer cannot compel the functions to change their systems and processes, which they control. This gap between formal and effective authority is why data officers frequently find their initiatives stalled: they have the formal mandate to set data standards and lack the effective power to make functions adopt them, and the gap is filled only by executive backing above the functional level that can compel adoption, which is precisely what data officers frequently lack.

Bridging the formal-effective authority gap is therefore the crux of making the settlement stick, and it requires something the organizational chart alone does not provide: active, sustained sponsorship from an authority above the functions, typically the chief executive or a similarly-placed leader, who is willing to compel the functions to adopt the settlement even over their resistance. Without such sponsorship, the data officer's formal authority is insufficient, and the settlement, which requires functions to change, does not happen. With it, the data officer becomes the agent of an executive mandate that the functions cannot ignore, and the settlement becomes possible. This is why the master data settlement is frequently described as requiring executive sponsorship, and why that sponsorship is not a nicety but a necessity: the settlement redistributes definitional authority, the functions resist the redistribution, and only an authority above the functions can compel it, so the settlement lives or dies on whether such an authority actively backs it. An enterprise that assigns the master data mission to a data officer without securing the executive sponsorship that gives the officer effective authority has set the mission up to stall, because the officer will have the formal mandate and lack the effective power, and the settlement will founder in the gap between them.

06 Centralized, federated, and the golden-record trap

The dominant technical approach to master data, the centralized golden record, embodies the misdiagnosis this article describes, and understanding why it so often fails illuminates the better alternative. Figure 5 contrasts the two governance models.

Two governance models, and why the golden record so often fails



Centralized MRD aims to create a single golden record, one version of the truth, clear ownership, and a single point of truth. Federated governance keeps a shared identity, local views, and governance across functions. The federated model is more complex, but it is the only model that can handle the complexity of the real world.

Figure 5. Two governance models. The centralized golden record gives one version of the truth but forces false consensus and destroys legitimate nuance; federated governance keeps a shared identity spine with governed local views, preserving valid differences at the cost of requiring real governance maturity.

The centralized approach builds a single golden record for each entity, one authoritative version of the customer, the product, the supplier, that all functions are meant to use. Its appeal is obvious: one version of the truth, clear ownership of that version, and an end to the inconsistency of multiple definitions. And for some entities and some purposes it works, because some entities truly do have a single correct definition that all functions can share. The trouble is that for the contested entities, the ones where functions legitimately differ, the centralized golden record does not resolve the difference; it suppresses it, choosing one definition and imposing it on functions whose correct definitions differ, which forces a false consensus that the suppressed functions cannot actually use.

This is the golden-record trap. The enterprise, seeking to end inconsistency, builds a single record that forces every function to the same definition, and in doing so it destroys the legitimate differences that the multiple definitions encoded. Finance's need to track the paying entity, marketing's need to track the individual, legal's need to track the hierarchy, these are real needs served by real differences in definition, and a golden record that collapses them into one definition serves whichever function's definition it chose and fails the others. The functions whose definitions were suppressed either abandon the golden record and revert to their own definitions, restoring the inconsistency the golden record was meant to end, or comply with the golden record and lose the ability to serve their legitimate purposes. Either way the golden record fails, because it tried to legislate away differences that were real, and real differences cannot be legislated away; they can only be suppressed, at the cost of the purposes they served, or accommodated, which the golden record does not do.

The federated alternative accommodates the differences rather than suppressing them, and it is the approach the constructive sections develop. Instead of a single golden record that forces one definition, federated governance maintains a shared identity spine, establishing that an entity exists and giving it a stable identifier, while allowing each function to keep its own governed view of that entity, linked to the shared identity. Finance's payer view, marketing's contact view, legal's hierarchy view all coexist, each serving its function's legitimate purpose, and all linked to the same underlying identity so that they can be reconciled. This preserves the valid nuance that the golden record destroys while making the definitions reconcilable through their shared identity, which is what the enterprise actually needs: not

one definition that serves one function and fails the rest, but multiple governed definitions that serve their functions and reconcile through a common spine. The federated approach demands more governance maturity than the centralized one, because governing multiple linked views is harder than imposing one record, which is why it is less commonly attempted. But it succeeds where the golden record fails, because it works with the legitimate multiplicity of definitions rather than against it, and the master data problem, correctly understood, is a problem of governing legitimate multiplicity, not of eliminating it.

It is worth naming a hybrid pattern that many mature enterprises settle into, because it captures the practical middle ground between the pure golden record and full federation. In this pattern, the enterprise consolidates the uncontested attributes of an entity, the ones all functions agree on, into a shared core, while federating the contested attributes, the ones functions define differently, into governed functional views. A customer's fundamental identity, name, and primary identifiers might be consolidated, because all functions agree on them, while the customer's status, value, and relationship, which functions define differently, remain federated. This hybrid consolidates where consolidation is legitimate and federates where federation is necessary, applying each approach to the attributes for which it is appropriate rather than choosing one approach for the whole entity. It is more nuanced than either pure approach and, for that reason, more likely to fit the actual structure of the problem, in which some attributes are truly shared and others are legitimately divergent, and the settlement should reflect that mixed reality rather than forcing the entity wholly into one model or the other.

07 Data mesh and governed multiplicity

The intellectual framework that most clearly legitimizes the federated approach, and that supplies the principles for governing multiplicity, is data mesh, and although it arrived wrapped in its own hype cycle, its core ideas directly address the master data problem. Figure 6 sets out its principles.

Data mesh: the framework that legitimizes governed multiplicity



Figure 6. Data mesh rests on four principles: domain ownership, data as a product, a self-serve platform, and federated computational governance. It explicitly rejects the single source of truth in favor of multiple connected models, validating governed multiplicity.

Data mesh, articulated by Zhamak Dehghani beginning around 2019 and developed in essays and a book, rests on four principles. Domain ownership holds that the domain that knows the data, that produces and understands it, should own it, rather than a central team that does not understand it owning it. Data as a product holds that data should be treated as a product, with defined consumers, quality standards, and a producer accountable for serving its consumers well. The self-serve platform provides shared infrastructure so that domains can publish and consume data without each building its own plumbing. And federated computational governance establishes global rules that apply across domains while leaving domains autonomy within those rules, with the governance implemented computationally, in the platform, rather than manually.

The principle most directly relevant to the master data problem is one that Dehghani states explicitly: data mesh rejects the single source of truth in favor of multiple connected models. This is precisely the governed-multiplicity approach the master data problem requires, stated as a general principle of data architecture. Rather than forcing all data to one authoritative version, which is the golden-record approach that fails, data mesh accepts that different domains will model the same concepts differently, for their own legitimate reasons, and focuses on connecting those models rather than collapsing them. Applied to the definition of a customer, this says exactly what the federated approach says: let each domain keep its own model of the customer, serving its own purpose, and connect the models rather than forcing them to one. Data mesh supplies the architectural legitimacy and the governing principles for the settlement this article recommends, from a source independent of the master data software vendors.

Honesty about data mesh requires acknowledging its hype cycle, which peaked around 2022 and passed into a trough by 2024, leaving organizations to keep the useful parts rather than adopt the whole movement wholesale, and only a minority have reached the governance maturity that federated computational governance demands. The irony, worth naming, is that data mesh requires exactly the governance maturity that centralized master data programs lacked and that their failures reflected: federated governance is harder than centralized, not easier, and an organization that could not sustain a centralized golden record may struggle more with federated governance, not less. So data mesh is not a magic solution that makes the master data problem easy; it is a framework that correctly diagnoses the

problem as one of governing multiplicity rather than eliminating it, and that supplies principles for doing so, while demanding a governance maturity that remains scarce. The value of data mesh to this article is its principled validation of governed multiplicity and its rejection of the single source of truth, which are correct, not a claim that adopting data mesh terminology solves a problem that remains, at its core, a hard political and organizational settlement requiring authority, tenure, and governance maturity that most organizations still lack.

08 Why this is now an AI problem

The master data problem is not new, and enterprises have tolerated it for decades, absorbing the reconciliation difficulties and the inconsistent reports as a cost of doing business. What has changed, and what makes the problem urgent now, is artificial intelligence, which raises the stakes of the definitional gap from an inconvenience to a barrier. Figure 7 shows why.

Why the master-data gap is now an AI problem, not just a reporting one

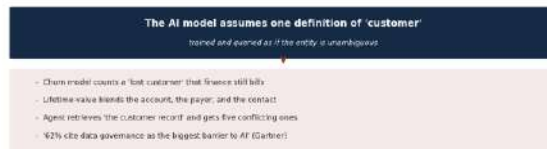


Figure 7. Why the master-data gap is now an AI problem. A model trained or queried as though 'customer' has one definition inherits the ambiguity it assumes away, and Gartner reports a large share of organizations cite data governance as the top barrier to AI.

An artificial-intelligence model, whether a predictive model or a generative system answering questions, operates on the data it is given and the definitions embedded in that data, and it assumes, because it must, that the entities it reasons about are well-defined. A churn model trained to predict which customers will leave assumes it knows what a customer is; a lifetime-value model assumes the customer whose value it computes is a single well-defined entity; a generative assistant asked about a customer assumes the customer it retrieves is unambiguous. When the underlying definition is in fact fragmented, the model inherits the fragmentation it assumed away, and its outputs are corrupted in ways that are hard to detect because the model presents them with the same confidence as any other output.

The failures are concrete. A churn model may count a customer as lost by one function's definition while finance still bills that customer under its definition, so the model's churn predictions do not match the financial reality and the retention efforts they drive are misdirected. A lifetime-value calculation may blend the account, the payer, and the contact, computing a value for an entity that does not coherently exist, and the resulting figure misinforms every decision that uses it. A generative assistant asked to retrieve the customer record may find five conflicting records, one per function's definition, and either present one arbitrarily, misleading the user, or present the conflict, revealing that the enterprise does not know what its own customer is. In each case the artificial intelligence does not create the definitional problem; it inherits it, and it surfaces it, because the model's need for a coherent definition exposes the incoherence that reporting had merely muddled through.

This is why data governance has become the leading barrier to artificial intelligence, cited by a majority of organizations in Gartner's finding, and why the master data problem, long tolerated, is now forcing itself onto the agenda. Enterprises rushing to deploy artificial intelligence discover that their models rest on definitions that do not cohere, that the customer the model reasons about is five different entities, and that the model cannot produce reliable outputs on an incoherent foundation. The artificial-intelligence ambition, in other words, has raised the cost of the unmade master data settlement from tolerable to prohibitive, because reporting could paper over the definitional gap while artificial intelligence cannot. An enterprise that wants reliable artificial intelligence must first make the master data settlement it has deferred for decades, because the models it wishes to deploy require the

coherent definitions the enterprise has never established, and no amount of modeling sophistication can compensate for a foundation in which the central entities are undefined. The artificial-intelligence era has made the master data mandate unavoidable, and the enterprises that treat their artificial-intelligence difficulties as modeling problems, when they are actually definitional ones, will find their models perpetually undermined by the incoherence beneath them.

The AI dimension has a further implication that raises the urgency still higher, concerning the speed and scale at which AI propagates the definitional error. A human analyst working with inconsistent customer definitions can apply judgment, noticing when a number seems wrong, caveating a reconciliation, flagging an ambiguity, so the definitional problem is partially contained by human awareness of it. An AI model applies no such judgment; it processes the data at scale, propagating the definitional error through every output it produces, without noticing the ambiguity a human would flag, and because it operates at machine scale and speed, it propagates the error far faster and more widely than human analysis would. The definitional problem that human analysis partially contained through awareness becomes, under AI, an error propagated at scale through systems that do not know it is there, which magnifies the consequences of the unmade settlement.

This magnification means that deploying AI on an unsettled master data foundation is not merely as risky as the old reporting problem but more so, because AI removes the human judgment that partially contained the problem and propagates the error at a scale and speed that human processes did not. An enterprise that tolerated its master data problem for decades, because human analysis muddled through it, may find that AI deployed on the same foundation produces errors at a scale that can no longer be muddled through, corrupting decisions and customer interactions faster than anyone can catch them. The AI era does not merely make the master data settlement worth making; it raises the cost of not making it, because AI amplifies the consequences of the definitional gap beyond what the enterprise experienced when humans, with their judgment and their awareness of the ambiguity, stood between the incoherent data and the decisions. The settlement that was optional in the reporting era becomes a precondition in the AI era, because the AI amplifies the unsettled problem to a scale at which its consequences can no longer be absorbed.

09 The identity spine and the governed view

Having diagnosed the problem as political and the golden record as the wrong technical response, this article can now set out the workable settlement, which separates a shared canonical identity from the function-specific views that legitimately differ. Figure 8 illustrates the model.

The recommended model: shared identity spine, governed federated views



Figure 8. The recommended model. A shared identity spine establishes that an entity exists and gives it a stable identifier; each function keeps its own governed view, account, payer, contact, seat, hierarchy, linked to that spine, preserving legitimate differences while making them reconcilable.

The core move is to distinguish two things that the golden record conflates: the identity of an entity, and the definition of that entity for a purpose. The identity is the fact that a particular customer exists, as a distinct thing in the world, referenced by a stable identifier that everyone agrees on. The definition is what that customer is for a given function, the account for sales, the payer for finance, and so on. The golden record tries to make the definition singular, forcing everyone to one definition, which fails because the definitions legitimately differ. The identity spine instead makes only the identity singular, establishing one agreed identifier for the entity, while allowing the definitions to remain multiple, each function keeping its own governed view linked to the shared identity.

The power of this separation is that it delivers what the enterprise actually needs, reconcilability, without demanding what it cannot have, a single definition. Because every function's view of the customer is linked to the same identity spine, the views can be reconciled: the enterprise can trace that sales's account, finance's payer, and marketing's contact all refer to the same underlying identity, and can therefore combine information across them correctly, answering cross-functional questions by joining the views through their shared identity. Yet because each function keeps its own view, no function loses the definition it needs; finance still has its payer, marketing still has its contact, and each serves its legitimate purpose. The separation gives the enterprise the reconciliation it needs for cross-functional work and the multiplicity it needs for functional work, which the golden record could not do because it forced a choice between them.

Implementing this requires establishing and governing the identity spine, which is a smaller and more tractable task than building a golden record, because it requires agreement only on identity, not on definition. The functions must agree on what counts as the same entity, so that they can link their views to a common identity, which is a real task requiring matching and de-duplication, but it is a far narrower agreement than agreeing on a single definition, because it leaves each function's definition intact and asks only that they agree on when two things are the same thing. Governing the spine means maintaining the identity resolution as entities are created, merged, and changed, ensuring that the links between the functional views and the spine stay correct. This is ongoing work, but it is bounded and technical in a way the political settlement over definitions is not, because the hard political question,

whose definition governs, is dissolved rather than answered: no function's definition governs, because each keeps its own, and the shared thing is only the identity, which is less contested because it advantages no function's definition over another's. The identity spine is the technical mechanism that makes the political settlement tractable, by reducing the shared commitment to identity, which functions can more readily agree on, and leaving definition federated, which is what functions actually need.

A practical question arises about how strict the identity spine must be, and the answer is that it must be strict about identity precisely because it is permissive about definition. The whole design rests on every functional view referencing the same underlying identity, so that the views can be reconciled through it, and this reconciliation works only if the identity resolution is reliable, if the enterprise can trust that two views pointing to the same identity really do refer to the same entity. If the identity spine is sloppy, merging entities that are distinct or failing to merge entities that are the same, the reconciliation it enables is corrupted, and the federated model loses the coherence that justifies it. So the identity spine demands rigor at exactly the point where the golden record demanded rigor, in matching and de-duplication, but it demands that rigor only about identity, not about definition, which is a narrower and more achievable target. The enterprise concentrates its data-quality effort on getting identity right, because identity is what the whole federated structure depends on, and accepts that definition is federated, which relieves it of the impossible task of getting a single definition right across functions that legitimately differ.

10 The vendor claims, and what software cannot do

The master data software market makes claims that a buyer, having understood the political nature of the problem, should evaluate with care, because the claims frequently promise that software will deliver what only a political settlement can, and the gap between the promise and what software can actually do is the source of much master data disappointment.

The vendors of master data management platforms sell the golden record: software that will consolidate scattered records into a single authoritative version and thereby, the claim goes, resolve the enterprise's master data problem. The software plainly does what it says at the technical level, matching, merging, and de-duplicating records, maintaining a consolidated store, and providing the tools to manage it, and these are real capabilities that a master data effort needs. The difficulty is not that the software fails to do what it claims technically; it is that doing what it claims technically does not resolve the problem, because the problem is political, and consolidating records into a golden record suppresses rather than settles the definitional disagreement, as the earlier sections argued. The software delivers the golden record and the golden record does not deliver the resolution, because the resolution required a political settlement the software cannot make.

A buyer should therefore read master data software claims with a clear understanding of the boundary between what software can and cannot do. Software can maintain an identity spine, resolving which records refer to the same entity and giving that entity a stable identifier, which is materially useful and is the technical foundation of the federated approach this article recommends. Software can maintain governed views, storing and managing each function's definition and linking them to the identity spine. Software can enforce governance rules, applying agreed policies computationally. What software cannot do is make the agreements that the governance enforces: it cannot decide whose definition governs the shared uses, cannot negotiate the settlement between functions, cannot supply the authority to make the settlement hold. Those are political acts, and a buyer who expects the software to perform them will be disappointed, because no software can, and the vendors who imply otherwise are selling a technical solution to a political problem.

The constructive reading is that software is necessary but not sufficient, and that its role is to implement a settlement that must be made politically first. An enterprise that makes the political settlement, deciding to adopt the federated model, agreeing on identity, establishing the governed views and their ownership, and then acquires software to implement that settlement, is using the software correctly, as the technical instrument of a political decision already made. An enterprise that acquires software first, expecting it to make the settlement, has the order backward and will find the software deployed and the problem unresolved, because it bought a tool to make a decision that only the organization can make. This publication has no stake in any vendor, and its counsel is that master data software is worth having once the political settlement is made and worthless as a substitute for making it, that a buyer should therefore make the settlement first and buy the software to implement it, and that a buyer should read every vendor claim that software will resolve the master data problem as eliding the political settlement that is the actual difficulty and that software, however capable, cannot perform.

11 The fairness case: some multiplicity is real

This article has argued for governed multiplicity over the single golden record, and fairness requires acknowledging the genuine case for consolidation and the real risks of the federated approach, because the truth is not that multiplicity is always right but that the choice requires judgment the article should not pretend away.

The strongest point in fairness is that not all multiplicity is legitimate, and some of it is simply mess that should be consolidated. This article has emphasized the legitimate differences between functional definitions, but many enterprises also carry illegitimate multiplicity, the same entity recorded inconsistently for no good reason, definitions that differ through accident and neglect rather than genuine functional need, records duplicated and diverged through poor data hygiene. This illegitimate multiplicity is real mess, and for it the golden-record instinct is correct: it should be consolidated, because there is no legitimate difference to preserve, only accidental inconsistency to eliminate. The skill is distinguishing the legitimate multiplicity, which the federated approach should preserve, from the illegitimate mess, which consolidation should eliminate, and an enterprise that treats all multiplicity as legitimate, refusing ever to consolidate, is as wrong as one that treats all multiplicity as mess, forcing everything to a golden record. The federated approach is right for the legitimate differences and wrong for the accidental ones, and the judgment about which is which is essential.

A second point in fairness is that forced consolidation, though this article has criticized it, is sometimes the right answer, particularly for entities that truly do have a single correct definition across all functions. Some master data entities are not contested: a product's fundamental identity, a supplier's legal registration, a location's physical existence may be things all functions can and should share a single definition of, and for these the golden record is appropriate and the federated overhead unnecessary. The federated approach's cost, the governance of multiple views, is justified only where the multiplicity is legitimate, and imposing it on entities that have a single correct definition adds complexity for no benefit. So the choice between consolidation and federation is itself a per-entity judgment: consolidate the entities with a single correct definition, federate the entities with legitimately multiple definitions, and do not apply either approach universally.

A third point in fairness is that the federated approach carries real risks that this article should not minimize. Federated governance is harder than centralized governance, demanding a governance maturity that many organizations lack, and an organization that attempts federation without that maturity may end up with the worst of both worlds: multiple definitions and no effective governance of them, which is just the original mess dignified with a fashionable name. Heavy master data programs are frequently over-engineered, and the federated approach can become its own form of over-engineering, an elaborate governance apparatus that consumes more than the problem warrants. And data mesh, the framework behind federation, has its own governance risks and its own hype-driven excesses. The honest synthesis is that the choice between consolidation and federation is a judgment requiring the enterprise to assess which of its entities have legitimate multiplicity and which do not, to gauge its own governance maturity candidly, and to apply the approach proportionate to the problem, neither forcing everything to a golden record nor federating everything into an ungovernable sprawl. This article argues

that governed multiplicity is the right answer for the contested entities that legitimately differ, which are the ones that cause the master data problem, not that federation is universally correct or that consolidation is always wrong.

12 Settle the definition, do not legislate it away

The governing principle for the master data problem follows from the diagnosis and can be stated as an instruction that captures the difference between the failing approach and the working one: settle the definition, do not legislate it away. Figure 9 sets out the resulting discipline.



Figure 9. Settling the definition without pretending the differences away: name the political owner, separate identity from views, enumerate the legitimate views, govern definitions as products, and start with the few entities that matter.

To legislate the definition away is what the golden record does: it decrees a single definition and suppresses the others, pretending the legitimate differences do not exist, and it fails because the differences are real and reassert themselves. To settle the definition is what the federated approach does: it acknowledges the legitimate differences, establishes a shared identity that makes them reconcilable, names an owner with authority over the settlement, and governs the multiple definitions as a managed system, working with the differences rather than pretending them away. The distinction is between suppression, which fails because it denies reality, and settlement, which succeeds because it accommodates reality, and the whole argument of this article reduces to preferring settlement over suppression.

Settlement has specific components that the discipline must include. It requires naming the political owner, the party with authority to make and sustain the settlement, and giving that owner the real authority and executive backing that the vanishing chief data officer lacked, because settlement requires authority and the settlement fails without it. It requires separating the canonical identity from the function-specific views, establishing the identity spine that makes the views reconcilable while leaving each function its own definition. It requires enumerating the legitimate views explicitly, listing which functions legitimately need a different definition and why, so that the multiplicity is deliberate and governed rather than accidental and unmanaged. It requires governing the definitions as products, with owners, consumers, quality standards, and change processes, treating each definition as a managed thing rather than an accident. And it requires starting with the few entities that matter, rather than attempting to settle every entity at once, which the next section develops.

The reframing from legislation to settlement changes what the enterprise attempts and how it succeeds. An enterprise attempting to legislate the definition, via a golden record that forces consensus, is attempting something that cannot succeed for the contested entities, because it denies their legitimate differences, and its failure is structural, not a matter of execution. An enterprise attempting to settle the definition, via a federated model that accommodates the differences, is attempting something that can

succeed, because it works with reality rather than against it, though it requires the authority, the governance maturity, and the sustained ownership that settlement demands. The choice between the two determines whether the master data effort is capable of success at all: legislation is structurally incapable of settling the contested entities, however well executed, while settlement is capable, if the enterprise supplies the authority and governance it requires. An enterprise that has understood this stops attempting to legislate its definitions into singularity and starts settling them into governed multiplicity, which is the difference between the master data programs that perpetually fail and the ones that eventually work.

13 A governance protocol, and a scoring rubric

The principles above combine into a governance protocol for the master data settlement, and a rubric for assessing whether a master data effort is set up to succeed or is repeating the golden-record error.

The protocol runs as follows. Assign definitional authority explicitly and politically, naming an owner with real authority and executive backing to make and sustain the settlement. Separate the canonical identity from the function-specific views, building a shared identity spine and letting each function keep its governed view linked to it. Enumerate the legitimate views for each contested entity, distinguishing legitimate functional differences from accidental mess, consolidating the mess and federating the legitimate differences. Govern the definitions as products, with owners, consumers, quality standards, and change processes. And start minimum-viable, settling the few entities that matter most rather than attempting the whole enterprise at once, which the next section develops.

A scoring rubric

The dimensions below distinguish a master data effort set up to settle from one set up to fail.

Dimension	Set up to settle	Set up to fail
Framing	Political settlement with tech support	Technical project owned by IT
Owner	Real authority, executive backing	Contested authority, short tenure
Identity vs definition	Separated: shared spine, local views	Conflated into one golden record
Multiplicity	Legitimate differences preserved	All differences suppressed
Governance	Definitions governed as products	Records consolidated, then unmanaged
Scope	Few entities that matter, first	Boil the ocean, all at once
Software role	Implements a settlement already made	Expected to make the settlement

An effort scoring in the left column is set up to settle: framed politically, owned with authority, separating identity from definition, preserving legitimate multiplicity, governing definitions as products, scoped to what matters, and using software to implement a decision already made. An effort scoring in the right column is set up to fail: framed technically, owned without authority, conflating identity and definition into a golden record, suppressing legitimate differences, and expecting software to make a settlement it cannot. The rubric does not make the master data settlement easy, which the politics prevents. It ensures the enterprise is attempting something that can succeed, a governed settlement, rather than something that cannot, a legislated golden record.

14 Minimum-viable governance, not the enterprise ocean

A final principle distinguishes master data efforts that make progress from those that collapse under their own ambition: start with minimum-viable governance of the few entities that matter, rather than attempting to govern every entity across the whole enterprise at once. The boil-the-ocean approach is a common and predictable way master data programs fail, and its avoidance is essential.

The boil-the-ocean impulse is understandable. The enterprise has master data problems across many entities, customers, products, suppliers, locations, and more, and the instinct is to solve them comprehensively, launching a program to govern all master data across all functions. But this comprehensive ambition is precisely what makes master data programs collapse, because it requires making the difficult political settlement for every entity simultaneously, negotiating with every function about every definition at once, which is a scope so large that it cannot be completed before the owner departs, the sponsorship wanes, or the organization loses patience. The comprehensive program attempts the multi-year political settlement across the entire entity landscape in one effort, and the effort is too large to succeed, so it produces years of activity and little settlement, and it becomes one of the failures the statistics count.

The minimum-viable alternative settles the few entities that matter most, deeply and completely, before extending to others. Most enterprises have a small number of master data entities that cause most of the pain, frequently the customer above all, and settling those few entities well, making the political settlement, establishing the identity spine and governed views, delivers most of the value while requiring a fraction of the scope. The enterprise settles the customer, demonstrably and completely, then applies the pattern to the next most important entity, and the next, extending the settlement incrementally as each is completed rather than attempting all at once. This incremental approach is more likely to succeed because each step is a tractable political settlement of a bounded scope, completable within an owner's tenure and a sponsor's patience, and because each completed settlement demonstrates value that sustains support for the next.

The minimum-viable principle also disciplines the governance itself, not just the scope, favoring the lightest governance that works over the most comprehensive governance imaginable. Heavy master data governance, elaborate committees, extensive documentation, exhaustive controls, is frequently over-engineered, consuming more effort than the problem warrants and creating a bureaucracy that impedes rather than enables. The minimum-viable approach asks what the least governance is that would actually settle the entity and keep it settled, and does that, rather than building the comprehensive governance apparatus that master data methodologies often prescribe. This is not an argument for inadequate governance, which fails, but for proportionate governance, which settles the entity with the least apparatus that suffices, because the goal is the settlement, not the governance, and governance beyond what the settlement requires is overhead that makes the effort heavier and more likely to collapse. An enterprise that settles its few most important entities with minimum-viable governance, incrementally and proportionately, makes the progress that the boil-the-ocean program does not, because it attempts what can be completed rather than what cannot, and it delivers the value of a settled customer, the entity that matters most, rather than the perpetual activity of a

comprehensive program that settles nothing. The master data mandate is met not by governing everything at once but by settling the entities that matter, one at a time, with the authority to make each settlement and the discipline to keep each governance proportionate to what the settlement requires.

The minimum-viable principle connects back to the political diagnosis in a way that closes the argument. Because the settlement is political, requiring authority to make and sustain, and because that authority is scarce and its holder's tenure is short, the settlement must be scoped to what the available authority can actually accomplish within the time it is available. A boil-the-ocean program presumes an authority and a tenure that the enterprise does not have, attempting a settlement so large that no realistic amount of authority could impose it before the owner departs. A minimum-viable program scopes the settlement to what the available authority can impose within the available tenure, settling the few entities that matter and thereby fitting the political reality that authority is limited and tenure is short. The minimum-viable approach is not merely a project-management preference; it is the scoping that the political diagnosis requires, matching the ambition of the settlement to the authority available to make it, and an enterprise that scopes its master data effort to its actual authority succeeds where one that scopes to an imagined comprehensive authority fails.

This closes the loop between diagnosis and prescription. The diagnosis is that master data management is a political settlement requiring authority and tenure that enterprises struggle to provide; the prescription follows directly, that enterprises should scope the settlement to the authority and tenure they actually have, settling the few entities that matter with the authority available rather than attempting the whole enterprise with authority they lack. Every element of the recommended approach, naming an owner with real authority, securing executive sponsorship, separating identity from definition to reduce what must be agreed, starting minimum-viable, reflects the same underlying recognition: that the binding constraint on master data success is political authority, and that the approach must be designed around that constraint rather than ignoring it. The enterprises that succeed at master data are the ones that recognize its political nature and design their approach around the scarcity of the authority it requires; the ones that fail are the ones that treat it as a technical project and are surprised when the technology, however capable, cannot supply the political settlement that the problem actually needs. The master data mandate is met by those who understand what kind of problem it is, and designed their response to fit.

15 Conclusion: name the owner, keep the views

The master data problem endures because it is misdiagnosed. An enterprise that cannot agree what a customer is, and whose reports and models founder on the disagreement, reaches for a technical solution, the master data platform and the golden record, because the problem presents as a technical one, inconsistent records, and technical solutions are what enterprises know how to buy. But the inconsistent records are the symptom, and the disease is the unmade political settlement about whose definition of a shared entity governs, which no software can make. The enterprise buys the platform, builds the golden record, and finds the problem unresolved, because it treated a political disease with a technical remedy, and the remedy, however well implemented, could not reach the disease.

The diagnosis this article offers is that master data management is a political settlement, not a technical project, and that its frequent failure follows from that misdiagnosis: from treating as a technology problem what is actually a question of authority over shared definitions, from assigning that question to a role, the chief data officer, configured with contested authority and short tenure, and from attempting to resolve it with a golden record that suppresses the legitimate differences between functions rather than settling them. The failures are structural consequences of the misdiagnosis, and they will recur as long as the misdiagnosis does, however good the software becomes, because software cannot supply what the problem requires, which is a political settlement made and sustained by authority.

What a clear-eyed enterprise does follows from the correct diagnosis and can be stated simply: name the owner, and keep the views. Name the owner means assigning the settlement to a party with real authority and executive backing to make it and sustain it, because a political settlement requires authority and fails without it, and the vanishing data officer with contested authority is not enough. Keep the views means adopting the federated model, a shared identity spine that makes the definitions reconcilable and governed function-specific views that preserve the legitimate differences, rather than the golden record that suppresses them, because the multiplicity of definitions is largely legitimate and must be accommodated, not legislated away. Around these two moves cluster the supporting disciplines: separating identity from definition, enumerating the legitimate views, governing definitions as products, starting with the few entities that matter, and using software to implement a settlement already made rather than expecting it to make one. The artificial-intelligence era has made this mandate unavoidable, because the models an enterprise now wishes to deploy require the coherent definitions it has deferred establishing for decades, and no modeling sophistication can compensate for a foundation in which the central entities are undefined. The enterprises that meet the mandate will do so not by buying better software but by making the political settlement the software cannot make, naming an owner with the authority to settle the definition and keeping the governed views that let the definition be both singular in identity and plural in meaning. What is a customer? The answer, correctly given, is that it is one identity and several governed views, and an enterprise that can give that answer has met the master data mandate, while one that insists on a single definition, or tolerates an unsettled multiplicity, has not.

16 Methodology, caveats, and sources

Methodology

- This article synthesizes analyst commentary on master data and governance, data-mesh literature, chief-data-officer tenure research, and reporting on data governance as a barrier to AI, current to mid-2026. Supply Chain Research is independent and accepts no payment from the software vendors or data-services providers discussed.
- The master data failure statistics vary widely and several originate with parties that sell data services. They are treated as directional, and their disagreement is analyzed as evidence rather than presented as fact.

Caveats

- The failure rates cited, ranging from roughly 75 to 90 percent, come from sources using inconsistent definitions of failure, several traceable to interested vendors. The precise figures should carry little weight; their uniform height and mutual inconsistency are the reliable signals.
- The chief-data-officer tenure figures are drawn from MIT Sloan research and industry surveys and represent averages across varied populations. Individual tenures vary widely, and the roles and their boundaries are evolving as the chief AI officer emerges.
- Data mesh is presented for its principled validation of governed multiplicity, not as an endorsement of the full movement or its terminology. It carries its own hype cycle and governance risks, noted in the text.
- The five-function account of the definition of a customer is a representative illustration, not a claim that every enterprise has exactly these five definitions. The number and nature of legitimate definitions vary by organization.
- Figures 1, 3, and 8 are conceptual illustrations of a structure and a recommended model, not measured data, and are labelled as such.
- The identity-spine and governed-view model is a design pattern, not a product. Its success depends on the political settlement and governance maturity discussed throughout, and it should be adapted to the specific entities, functions, and constraints of the adopting organization rather than applied as a template.

Sources

- [1] Dataversity. [Common master data management pitfalls, including the widely-cited failure statistics \(flag: D&B-sourced\)](#).
- [2] Start with Data. [Why MDM projects fail: data-management problems and solutions \(flag: consultancy\)](#).
- [3] MIT Sloan. [Chief data officers don't stay in their roles long: here's why](#).
- [4] Becker's Health IT. [Average chief data officer tenure just 2.5 years: seven things to know](#).
- [5] Dehghani, Z. [Data Mesh Principles and Logical Architecture \(martinfowler.com\)](#).
- [6] Dehghani, Z. [Data Mesh: Delivering Data-Driven Value at Scale \(O'Reilly, 2022\)](#).
- [7] Gartner. [On master data software not filling the MDM gap, and data governance as an AI barrier \(analyst commentary\)](#).
- [8] Medium (K. Karya). [Your platform is governed, your numbers still don't match: a master data problem](#).

Additional context drawn from analyst commentary on governance, from data-mesh literature, and from reporting on data governance as a barrier to AI. Failure statistics originating with interested parties are identified as such. This article is analysis, not data-governance or procurement advice, and its conclusions should be validated against your own circumstances before any decision.

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