



BEYOND THE RETROFIT

Why Data Governance Must Be Designed Into Delivery, Not Added Afterwards

A Practitioner's Perspective on Governance-by-Design in Banking Risk-Data Programmes

Giovanni Leonardi

November 2013

Contents

1. Executive Summary	3
2. The Retrofit Reflex	3
3. Why Late Governance Fails Structurally	4
4. What the New Principles Actually Changed	5
5. What "Designed In" Actually Means	6
6. The Six Dimensions, and Where Each Is Built In	6
7. The Objection: Cost, Speed, and the Appetite to Wait	7
8. Closing Reflection	8

Why Data Governance Must Be Designed Into Delivery, Not Added Afterwards

Transformation · No. 222

Originally written November 2013 | Re-edited and re-rendered for the WhereBeyond Edition, 2025

“Governance is not a layer you apply to finished data; it is a property of the way the data was built.”

1. Executive Summary

The prevailing habit in banking data programmes is to build first and govern later: stand up the platforms, wire the feeds, produce the reports, and only afterwards ask who owns the data, whether it can be trusted, and where it came from. This essay argues that the habit has become untenable. The Basel Committee's principles for effective risk data aggregation and risk reporting, issued at the start of this year, have made explicit what experienced practitioners already knew in their bones: the quality of a firm's risk data is a board-level concern, and it cannot be retrofitted onto an architecture that was never designed to carry it.

The argument runs in three movements. First, that late governance fails not through negligence but for structural reasons — ownership, lineage and quality are decisions about how data is produced, and by the time the data exists those decisions have already been made badly. Second, that the new supervisory expectations are best read not as a reporting obligation but as a demand for accountability that reaches back into design. Third, that governance-by-design is a concrete discipline, not a slogan: it lives in programme plans, in product and data-model design, and above all in acceptance criteria, and it can be specified precisely across six dimensions — ownership, quality, lineage, access, retention and accountability.

The conclusion is uncomfortable for programmes already in flight: you cannot inspect governance into a delivery at the end, any more than you can inspect quality into a manufactured product on the loading dock. It has to be built in from the first design decision, or it is not really there at all.

2. The Retrofit Reflex

The pattern recurs with a regularity that has stopped surprising me. A programme is chartered to deliver a new risk platform, a consolidated data warehouse, or a regulatory reporting capability. The plan is organised around movement and mechanism: source systems, extract and load, transformation, a target model, a reporting layer. Governance appears, if it appears at all, as a workstream near the end — a data quality

tollgate before go-live, a stewardship model to be defined once the data is flowing, a lineage exercise scheduled for a later phase that a candid programme manager knows will be the first thing sacrificed when the timeline compresses.

The reasoning behind the reflex is understandable. Governance feels like overhead against a delivery date. It produces no demonstrable output in a steering meeting. It is easier to point at a populated table than at a defined ownership model. And there is a seductive logic to the sequence: surely we must have the data before we can govern it. So governance is deferred, and deferral hardens into omission.

What this reasoning misses is that governance is not an activity you perform on data once it exists. It is a set of decisions about how the data comes to exist in the first place — who is accountable for a given figure, what definition it must satisfy, what it is allowed to be derived from, how far back its provenance can be traced. Those decisions are made, explicitly or by default, at the moment the data is designed and built. To defer governance is not to postpone it. It is to make every one of those decisions badly, silently, and then to pay to unpick them later.

Deferring governance does not postpone the decisions it governs. It makes them anyway — silently, by default, and almost always badly — and then charges the programme a second time to reverse them.

3. Why Late Governance Fails Structurally

It is tempting to treat the retrofit problem as one of discipline: if only programmes were more rigorous, they would govern earlier. But the deeper truth is that late governance fails for reasons that no amount of diligence overcomes. Three of the governance dimensions in particular resist being added afterwards.

Consider lineage. To know where a number came from — through which transformations, from which source, under which business rule — you must capture that provenance as the data moves. If the pipelines were built without instrumenting lineage, the information is not merely undocumented; in a meaningful sense it no longer exists. Reconstructing it after the fact means reverse-engineering the intent of code and mappings that may have been written by people who have since moved on, and the result is an approximation dressed up as a record. A regulator asking a firm to demonstrate the provenance of a capital number will not be satisfied by an archaeologist's best guess.

Consider ownership. The value of a data owner is that they are accountable for definitions and quality before the data is used in anger — they adjudicate what a counterparty exposure means, they resolve the conflict between two systems that both claim to hold the golden record. Appoint an owner after the platform is live and you have given them a *fait accompli*: a model already built on assumptions they never sanctioned, definitions already hardcoded into transformations, conflicts already resolved by whichever developer got there first. The owner inherits a house they did not design and are now asked to certify.

Consider quality. Data quality is often imagined as a filter applied near the point of consumption — a set of checks that flag bad records before a report is run. But most quality problems are born upstream, at the point of capture or in a transformation that quietly drops or distorts a field. A quality regime bolted on at the reporting layer can detect that something is wrong; it can rarely prevent it, and it can almost never explain it. Quality that is designed in — defined at source, asserted at each handover, measured against an agreed standard — prevents defects. Quality that is added on merely counts them.

The common thread is that these are not properties of the finished data. They are properties of the process that produced it. And you cannot add a property to a process that has already run.

“You cannot inspect governance into a delivery at the end, any more than you can inspect quality into a product once it has left the line.”

4. What the New Principles Actually Changed

Much of the commentary on the risk-data aggregation principles reads them as an enhanced reporting standard — a demand for faster, more complete, more granular risk reports, particularly under stress. That reading is not wrong, but it is shallow, and a programme that takes only that message will build the wrong thing.

The more consequential shift is one of accountability and its direction of travel. The principles insist that a firm must be able to aggregate risk data accurately and reliably, across the group, on demand and at speed, including in a crisis — and, critically, that the board and senior management are answerable for the capability. That last point is what reaches back into design. If the board is accountable for the reliability of risk data, then reliability cannot be a downstream property discovered at reporting time. It must be an engineered property of the whole chain, from the systems that originate the data to the reports that present it.

The timing is not accidental. The financial crisis of five years ago exposed, among much else, that a number of firms could not answer basic questions about their own exposures quickly enough to act — not because the data did not exist somewhere, but because it could not be aggregated with confidence across fragmented systems and inconsistent definitions. The supervisory response is, in effect, a judgement that risk-data capability is part of a firm's safety and soundness. Read in that light, the principles are not asking for better reports. They are asking for a firm whose data is governed by construction — where accuracy, completeness, timeliness and adaptability are designed in because the alternative proved dangerous.

This reframing matters for how a programme is scoped. A reporting-led reading produces a programme that optimises the last mile — the aggregation and presentation

layer — and leaves the tangle of source definitions and undocumented transformations beneath it untouched. An accountability-led reading produces a programme that treats the whole chain as in scope, because the board's answerability does not stop at the reporting layer.

5. What "Designed In" Actually Means

Governance-by-design is easy to assert and easy to hollow out. To be real, it has to change three specific artefacts that every serious programme already produces. If it does not change these, it is decoration.

The programme plan. Governance activities must appear in the plan as dependencies of delivery, not as a parallel workstream that can be descoped without consequence. Defining ownership for a data domain is a predecessor to building the model for that domain, not a successor. Agreeing the authoritative definition of a critical data element is a predecessor to writing the transformation that produces it. When governance sits on the critical path — when you genuinely cannot proceed to build without it — it stops being optional. When it sits alongside, it is the first thing cut.

Product and data-model design. The design of the target model is where most governance is silently decided: which entity is authoritative, how a critical element is defined, what is allowed to derive from what, where the single trusted version of a figure lives. To design in governance is to make these decisions deliberately and with the accountable owner in the room, rather than letting them fall out of a developer's local choices. A data model designed with ownership and lineage in mind looks different from one designed only for query performance — it carries provenance, it distinguishes authoritative sources from convenient ones, it makes definitions explicit rather than implied.

Acceptance criteria. This is the sharpest lever and the most neglected. If a deliverable is accepted purely on functional grounds — the report runs, the numbers reconcile to a total — then governance has no teeth, whatever the plan says. If acceptance requires that every critical data element has a named owner, an agreed definition, demonstrable lineage to source, and a measured quality score against a defined threshold, then governance becomes a condition of done. Acceptance criteria are where governance either bites or evaporates. A programme that wants governance-by-design and does not change its definition of done is not serious.

*"Governance is not a layer you apply to finished data;
it is a property of the way the data was built."*

6. The Six Dimensions, and Where Each Is Built In

It helps to be concrete about what is being designed in. Across the programmes I have observed, effective data governance resolves into six dimensions, each of which has a natural point in the lifecycle where it must be settled. Left to drift past that point, each becomes a retrofit.

Dimensi on	What it settles	Where it must be designed in
Ownersh ip	Who is accountable for definition and quality	Domain design, before modelling begins
Quality	The standard a data element must meet, and how it is measured	At source and at every handover, not at the reporting layer
Lineage	The traceable path from source to report	Instrumented into the pipelines as they are built
Access	Who may see and use the data, under what basis	Into the model and platform design, not as a later security overlay
Retentio n	How long data is kept and when it is disposed	Into the data model and storage design from the outset
Accounta bility	How ownership and quality are governed and escalated	Into the operating model that goes live with the platform

Two of these deserve a further word because they are the ones most often left until it is too late.

Access is habitually treated as a security concern to be layered on before go-live. But entitlement is a governance decision — it depends on who owns the data and on what basis it may legitimately be used — and when it is bolted on late it tends to be crude, permitting too much because fine-grained control was never designed into the model. Designing access in means the model itself understands the difference between data that may flow freely and data that must be constrained.

Accountability — the operating model of stewardship, the forums where quality is reviewed, the path by which a disputed definition is escalated and resolved — is the dimension most likely to be promised and least likely to exist on the day the platform goes live. A platform delivered without its governance operating model is an orphan: technically complete, accountably hollow. The operating model must be designed and stood up so that it is running as the data starts to flow, not conceived afterwards as an afterthought.

7. The Objection: Cost, Speed, and the Appetite to Wait

The honest objection to all of this is that it is slower and more expensive at the front, and that programmes are not rewarded for front-loading invisible work. A sponsor who has committed to a delivery date sees governance-by-design as a tax on velocity — more analysis before anything is built, more conditions on acceptance, more people in the room when the model is designed.

The objection deserves a real answer rather than a pious one. The answer is that the cost does not disappear when governance is deferred; it moves, and it grows in transit. The definitional conflict not resolved at design time is resolved later as a production incident, a reconciliation break, a report that two parts of the firm cannot agree on. The lineage not captured as the pipeline was built is reconstructed later, at far greater cost, under regulatory time pressure, and with less confidence. The owner not appointed early is appointed in the middle of a remediation, inheriting problems they must now certify. Deferred governance is not cheaper. It is borrowed at a punitive rate of interest, and the new principles have shortened the term of the loan.

There is also a quieter cost that rarely makes it into a business case: the erosion of trust. Once a set of numbers has been shown to be wrong, or unexplainable, the business stops believing the platform, and belief is expensive to rebuild. Data that cannot be trusted is not used, and data that is not used returns none of the value that justified building it. Governance-by-design is, in this sense, not a control overhead but the precondition for the programme delivering any benefit at all.

8. Closing Reflection

The temptation, reading the new principles, is to treat them as a compliance problem to be managed — a set of boxes to be evidenced before a supervisory deadline. That would be a mistake, and an expensive one. What the principles articulate is a standard that good practitioners were already reaching toward: that data on which serious decisions rest must be governed by construction, not by inspection.

The programmes that will meet the moment are not the ones that add a governance workstream to an existing plan. They are the ones that let governance reshape the plan — that put ownership before modelling, definition before transformation, and demonstrable lineage and measured quality into the very definition of a completed deliverable. This is harder at the start and far cheaper across the life of the thing. More than that, it is the only approach that produces data a board can actually stand behind, which is, after all, what is now being asked of them.

Build the governance in, or accept that what you have built is not yet trustworthy — and that the bill for making it so has not been avoided, only deferred, at interest.