The Provenance Method

A Guide to Building Data Control Frameworks That Perform Under Scrutiny

Introduction: The Gap Between Visible Data Quality and Genuine Assurance

Why the gap exists

Despite the proliferation of data governance frameworks across the insurance industry, most share a common gap. Organisations have invested heavily in the structural architecture of governance - establishing data councils, appointing data owners and designating stewards responsible for defined domains. These elements are necessary, but they represent only the organisational scaffolding of governance, not its operational substance.

What is conspicuously absent from the majority of frameworks is any rigorous treatment of the granular controls that actually give effect to governance intent: The question of whether controls are adequate - appropriately designed, consistently applied and regularly tested - is rarely asked, let alone systematically answered.

This gap tends to become apparent at precisely the moments when genuine assurance matters most: an **audit,** a **regulatory review**, or a **data quality incident** that existing controls should have caught.

The difference between the appearance of control and genuine, auditable assurance lies in how comprehensively your framework has been designed. That means understanding not just what your data is, but how it flows - across systems, through business processes and into the use cases where its quality actually matters.

It means identifying the risks inherent within that flow with enough precision to assess their likely impact and it requires building a framework of controls that addresses those risks systematically, with the right balance of preventative and detective measures, working in concert across the entire data journey.

Done properly, this provides something that a set of Data Quality Indicators s alone never can: assurance that is **proactive**, **auditable** and genuinely **proportionate** to the risks your organisation faces.

The Provenance Framework and Capability Maturity: Understanding Where They Fit

If your organisation has undertaken a data management capability assessment - whether through DCAM or a similar maturity model -you will already have a picture of where your data quality controls capability sits relative to a defined standard. That is genuinely valuable work. It tells you what the problem is and how significant it is relative to peers and best practice benchmarks.

What it does not tell you is how to fix it at the level that actually matters: the specific controls that are absent or inadequate within your critical data flows and the documented risk analysis that should underpin them.

The Challenge

Getting these aspects right is harder than it sounds. Mapping data flows at a level of detail sufficient to support meaningful risk identification is a significant undertaking. It requires input

from business, IT, and risk functions that do not always speak the same language or share the same priorities.

It requires honest assessment of where existing controls are genuinely effective and where they are not - a judgement that is difficult to make objectively from the inside.

Lastly, it requires a clear methodology to translate all of that into a framework that is robust in design, workable in practice and defensible under scrutiny.

These are not insurmountable challenges. But they are the reason why organisations that attempt to build or overhaul their Data Control Frameworks without a structured approach frequently find themselves with something that falls short of what they need.

This guide sets out that structured approach - and what it takes to get there.

The Provenance Method

A structured approach to building data control frameworks that deliver genuine, auditable assurance over critical data flows

From Data Flow To Genuine Assurance

Map

Understand The Complete Journey

- Start at the use case, not the source – quality requirements are defined by how data is consumed
- Trace horizontal lineage across systems and vertical lineage through transformations
- Surface shadow processes and undocumented manual interventions
- Document existing controls at each step in the flow

Assess

Identify And Evaluate Risk

- Identify inherent risks at each step with precision – specific failure modes, not general categories
- Assess likelihood and impact grounded in the actual flow and use-case
- Evaluate effectiveness of existing controls objectively
- Establish residual risk position appetite to identify action required

Control

Build Proportionate Assurance

- Design preventative and detective controls that work in concert across the flow
- Balance primary controls targeting specific risks with compensating controls
- Write precise control narratives – unambiguous, repeatable and auditable
- Ensure the framework is risk proportionate and workable in practice

Starting Point	Measured Against	Produces
Inherent Risk	**Control Effectiveness**	**Residual Risk**

THE OUTCOME

Assurance that is proactive, auditable and genuinely proportionate to the risks your organisation faces

Performing under regulatory and audit scrutiny and at the moments when genuine assurance matters most

Identifying Your Critical Data Flows

Before you begin mapping flows, identifying risks and reviewing control adequacy, you need to answer a more fundamental question: ***which flows warrant the most rigorous treatment?***

Not all data flows carry equal risk and attempting to map and control everything with the same level of depth is neither practical nor necessary. The organisations that manage data quality most effectively are not those who try to control everything uniformly - they are the ones that have made deliberate, defensible decisions about where their most material exposures lie and concentrated their effort accordingly.

This section sets out how to make those decisions systematically.

Start With Your Non-Negotiables: Regulatory and Financial Returns

The most natural starting point for identifying critical flows is the category of outputs where data failure carries the most severe external consequences. These are your regulatory and financial returns: the submissions, disclosures and reports that are made to regulators, external auditors, or financial counterparties and where inaccuracy is not simply an operational problem but a compliance, legal, or reputational one.

For most organisations, this category will include some combination of prudential regulatory returns, financial statements and associated disclosures, tax filings, statutory and board-level reports and submissions to industry bodies or market infrastructure. The specific set will vary by jurisdiction, but the defining characteristic is consistent: these are outputs where the organisation has made a formal commitment to accuracy, where external parties are relying on that commitment and where the consequences of falling short are not discretionary.

Extend to Key Data-Dependent Processes

Beyond regulatory and financial outputs, most organisations have a broader set of business processes where data quality is genuinely material to outcomes, even if the consequences of failure are internal rather than regulatory. These warrant the same rigour of identification, though the criteria for inclusion are different.

The question to ask of any process is: ***if the data underpinning this process were materially inaccurate, incomplete or untimely, what would actually happen?***

Some processes are genuinely tolerant of imperfection - the impact of a data quality issue would be minor, correctable, and contained. Others are not. The processes that fall into the second category are the ones that belong on your critical list.

In practice, the processes most likely to meet this threshold share one or more of the following characteristics.

- They involve decisions with significant financial consequences – in insurance, that will be pricing, underwriting, credit assessment and reserving - where poor data translates directly into poor outcomes and potentially material loss.

- They involve customer-facing activity where data errors generate complaints, regulatory attention, or reputational damage.

- They rely on data to trigger automated or semi-automated actions at scale, where a single upstream error can propagate widely before it is detected. Or they feed other

critical processes downstream, acting as a data source whose integrity is a dependency for outputs that are themselves non-negotiable.

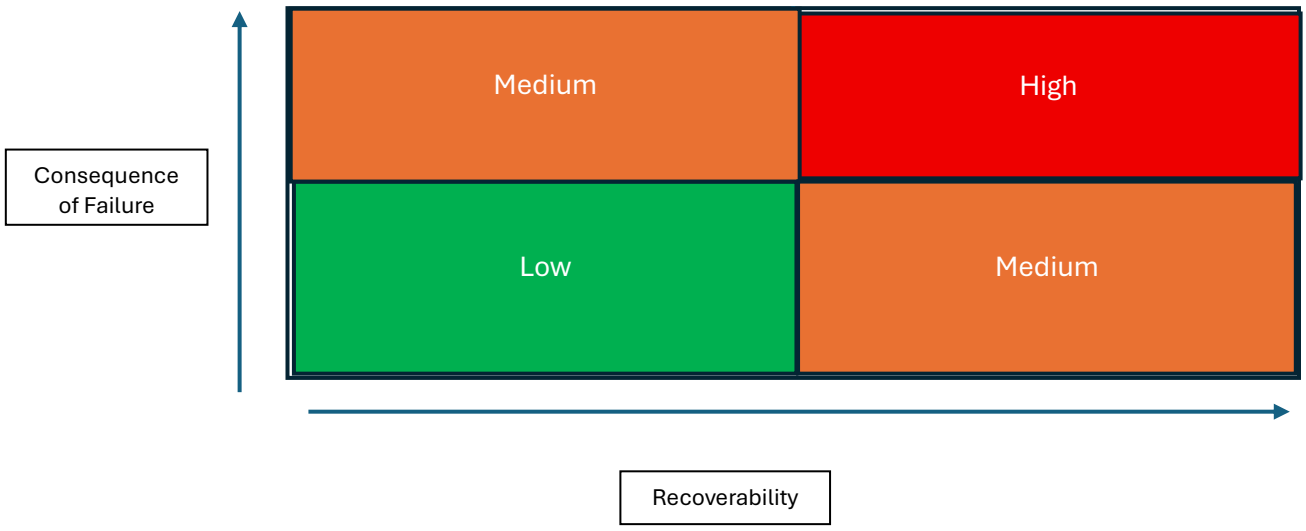
Identifying these processes requires genuine cross-functional engagement. Business teams will have direct visibility of which processes are most consequential in practice. Risk and compliance functions will have a view on where regulatory and conduct risk concentrations lie. IT and data teams will understand where systemic data dependencies exist that may not be obvious from the business perspective alone. None of these perspectives is sufficient on its own.

Assessing Degrees of Criticality

Identifying your critical flows is necessary but not sufficient. Within your critical set, there will be meaningful variation in how much scrutiny and control investment each flow warrants and your framework needs to reflect that variation rather than treating all critical flows as equivalent.

A structured approach to assessing degrees of criticality uses two dimensions:

- Consequence of Failure
- Recoverability



Consequence of failure considers what would actually happen if a material data quality issue occurred in this flow and went undetected through to the end use case. This is not simply a question of whether the output is important in the abstract - it is a question of the realistic worst-case outcome. At the severe end, this means enforcement action, financial restatement, or significant customer detriment. At the moderate end, it means operational disruption, management effort and reputational friction. At the lower end, it means an issue that is annoying and requires correction, but is ultimately contained. Be honest about where each flow sits; there is a natural tendency to grade conservatively, which produces a flat picture that obscures genuine differences in exposure.

Recoverability considers how quickly and completely a data quality issue in this flow can be identified and corrected once it has occurred. Some flows feed outputs that can be amended or resubmitted if an error is caught - the window for detection is meaningful and corrections are feasible. Others feed outputs that are effectively irreversible once submitted, or where

corrections are possible but carry their own regulatory or reputational cost. Flows with low recoverability warrant stronger preventative controls and earlier detective intervention than flows where timely correction is straightforward.

Plotting your critical flows across these two dimensions gives you a defensible basis for prioritising where to invest most heavily in your mapping and control design. Flows with severe consequences and low recoverability represent your highest-priority tier and should receive the deepest treatment. Flows with moderate consequences and reasonable recoverability represent an important but second-tier, where a rigorous but proportionate approach is appropriate. The objective is not to deprioritise any critical flow, but to ensure that the level of scrutiny applied to each one is proportionate to the genuine risk it carries.

Documenting Your Criticality Assessment

The output of this exercise should be a documented register of your critical flows, with the rationale for inclusion and the assessed degree of criticality recorded for each one. This serves two purposes beyond the immediate planning value.

1. It creates an audit trail that demonstrates your identification of critical flows was systematic and evidence-based, rather than a matter of assumption or convention. When a regulator or internal auditor asks why a particular flow received the level of control attention it did - or did not - you have a defensible answer.

2. It creates a living reference point. Criticality is not static. New regulatory requirements emerge. Business processes evolve. New data dependencies develop as systems and operating models change. The register should be reviewed at least annually, and whenever a significant business change occurs, to ensure that your control investment continues to follow your actual risk profile rather than a historical one.

With your critical flows identified and their degrees of criticality assessed, you have the prioritisation logic that should govern your entire mapping and control design exercise. Every subsequent decision - where to map in most depth, where to invest most heavily in controls, where to apply the most rigorous narrative standards - flows from the work done here.

Mapping Key Data Flows

Mapping your data flows is the analytical foundation of everything that follows. Without a sufficiently detailed and accurate picture of how your data moves - across systems, through business processes and into the use cases where its quality actually matters - any risk identification exercise you conduct will have blind spots.

Those blind spots in your risk identification mean blind spots in your Control Framework.

It is also, in our experience, the stage that organisations most consistently underestimate. The output looks straightforward. Getting there rarely is.

Start at the End: Understanding How Data is Consumed

The most important - and most frequently overlooked - principle in data flow mapping is this: start at the use case; not the source.

Many organisations approach flow mapping source-first, tracing data forward from where it originates. This produces technically accurate lineage but misses the point. The quality requirements that matter are determined entirely by how the data is ultimately used.

Starting from the source means you risk investing significant control effort in the wrong places - over-controlling low-stakes data while leaving genuine vulnerabilities unaddressed upstream of your most critical outputs.

Starting from the use case means every decision you make about where to focus your mapping effort and ultimately your controls, is anchored to what actually matters. For data feeding a regulatory return, near-complete accuracy is non-negotiable. For data used to derive statistical parameters in a model, the tolerance for imperfection maybe higher. That distinction should drive your entire approach and you can only make it clearly if you begin at the end.

Example: The Solvency Capital Requirement number for insurance carriers is more influenced by the processes that occur on the cusp of the Internal Model, or within it, than those further upstream. Suppose 5% of the individual case reserves feeding the capital model are wrong, by the time those records have been aggregated and actuaries have fitted a curve, the noise has been substantially reduced to the point of immateriality. However, a capital modeller making an invalid judgement on the parameterisation of the model could result in material swings of the end-capital number. That's where the risk is concentrated. Therefore it makes more sense to devote more time to the lineage components and processes at that point.

Assembling the Components of a Meaningful Flow

Once you understand how your data is consumed, you can build a flow with enough depth to support genuine risk identification. That flow has three components and understanding how they relate to one another is as important as understanding each one individually.

- ***Business Process Understanding*** is the foundation. Data does not exist in isolation - it is created, modified, and moved by people operating within processes. A technically accurate picture of system-to-system data movement tells you how data travels, but

without understanding the business processes that generate and act on that data, you cannot understand why it moves the way it does, where human judgment introduces variability, or where process breakdowns create quality risks that no system validation will catch.

This layer needs to capture not just the formal, documented process but the reality of how work is actually done. Shadow processes - workarounds that have evolved, manual interventions that are widely practised but nowhere documented - are frequently where the most material data quality risks reside. They will not appear in a system diagram. The only way to surface them is through direct conversation with the people doing the work.

- ***Technical Lineage*** sits on top of that business process understanding. You need to track two dimensions: **horizontal** lineage, which follows data as it passes from system to system; and **vertical** lineage, which tracks how data is transformed at each stage.

  The distinction matters because each dimension carries different risk profiles. Horizontal movement creates risks of loss and corruption - data that fails to transfer completely, or that is altered in transit in ways that are not immediately visible. Vertical transformations create a subtler but equally significant category of risk: a field that is aggregated, recalculated or re-coded may lose the granularity that a downstream use case depends on, or introduce errors that are difficult to trace back to their origin. Understanding both dimensions is essential to identifying where your controls need to sit.

- ***Controls Documentation*** completes the picture. As you map each business process, you need to capture what controls currently exist within it - where they sit, what they cover and who is responsible for operating them. This is not simply an administrative exercise. It is the information that allows you to move from inherent risk to residual risk: understanding not just where your vulnerabilities lie, but how much genuine protection your existing controls actually provide.

To assemble these components, you will need to speak to your Business Analysts, IT teams and the Risk and Compliance departments.

These conversations will enable you to build a flow similar to the sample one below.

Process Steps				Controls
OperationType	**Operation**	**Step No**	**Step Description**	**Controls**
Manual	Set up of Reinsurance Policy	**1**	The Policy Technician uses the information in the Underwriters Reinsurance Instructions to record the policy in the Reinsurance booking system.	A 100% QA check takes place over the set-up of the facultative cover in the booking system. This includes: 1. Checking the Underwriter Reinsurance Instructions to ensure that the facultative cover is linked to the correct direct policy and that the excess point is recorded corrrectly. 2. Checking that the inception and expiry dates on the facultative entry match those in respect of the direct policy on the inwards policy registration system.
Manual	Inward Claim proceessing	2	The claims adjuster checks the details of the inwards claim against the policy to determine the claims validity. If the claim is deemed as valid, the adjuster initiates the payment process.	A monthly report is run to check whether any direct policies with facultative cover attached have had a paid claim but no reinsurance recovery initiated. The technician checks each entry and must document the reason for the lack of a recovery.

A Note on Where Mapping Exercises Typically Stall

Producing a flow with the depth illustrated above requires meaningful input from functions that do not always speak the same language or share the same priorities. Business teams understand process but may lack visibility of system architecture. IT teams understand lineage but may lack the business context that gives it meaning.

Without active effort to bridge these perspectives, the result is typically a flow that is technically accurate but analytically incomplete.

Documentation quality presents a further challenge. In most organisations, process documentation is uneven, and where gaps exist, the temptation is to fill them with assumptions. That temptation should be resisted - an assumption documented as fact in your data flow will propagate silently into your risk identification and ultimately into your Control Framework.

Finally, assessing the effectiveness of existing controls from the inside is genuinely difficult. Controls that are technically present but inconsistently operated, or that were designed for a process that has since evolved, are easy to overlook when you are embedded in the environment they are meant to govern.

These are not reasons to avoid the exercise. They are reasons to approach it with the right structure, the right cross-functional engagement, and - in many cases - an independent and objective perspective.

Identifying Inherent and Residual Risks within the Data Flow

Completing a thorough data flow mapping exercise is a significant achievement. You have documented your business processes, traced your technical lineage and built a clear picture of how your data moves from source to use case. But the flow itself is not the destination - it is the analytical foundation for what comes next.

The breakdown of steps within your flow is what enables you to identify the risks inherent within it. This is where the real work of data quality risk management begins.

Inherent Risk

Inherent risks are those that exist within your data flow independently of any controls - the vulnerabilities that are a function of the flow's complexity, its touchpoints and the nature of the processes operating within it.

It is important that each inherent risk is identified and articulated with sufficient precision. Vague risk statements - "data may be inaccurate" or "systems may not align" - are of limited analytical value. The more clearly you can describe what could go wrong, where in the flow it could go wrong and why, the more useful your risk identification becomes as a basis for everything that follows. The example below illustrates the kind of precision to aim for.

Process Steps				Inherent Risk				Controls
OperationType	**Operation**	**Step No**	**Step Description**	**Risk Description**	**Impact**	**Likelihood**	**Score**	**Controls**
Manual	Set up of Reinsurance Policy	**1**	The Policy Technician uses the information in the Underwriters Reinsurance Instructions to record the policy in the Reinsurance booking system.	The data may not be transcribed correctly.	High	High		A 100% QA check takes place over the set-up of the facultative cover in the booking system. This includes: 1. Checking the Underwriter Reinsurance Instructions to ensure that the facultative cover is linked to the correct direct policy and that the excess point is recorded corrrectly. 2. Checking that the inception and expiry dates on the facultative entry match those in respect of the direct policy on the inwards policy registration system.
Manual	Inward Claim proceessing	2	The claims adjuster checks the details of the inwards claim against the policy to determine the claims validity. If the claim is deemed as valid, the adjuster initiates the payment process.	The inwards claim may not trigger the reinsurance recovery.	High	Medium		A monthly report is run to check whether any direct policies with facultative cover attached have had a paid claim but no reinsurance recovery initiated. The technician checks each entry and must document the reason for the lack of a recovery.

Assessing Extent of Risk

Once your inherent risks are identified, you need to assess their extent - the combination of likelihood and impact that determines how seriously each risk warrants being taken.

The same risk can have very different implications depending on where it sits in the flow and which use case it ultimately affects - a data quality issue upstream of a regulatory submission carries considerably more weight than the same issue upstream of an internal management report.

It is sufficient to grade likelihood and impact as High, Medium or Low, assigning numerical values as per the key in the exhibit. What matters is that the assessment is honest and grounded in your actual understanding of the flow, rather than defaulting to conservative gradings that obscure where your real exposure lies.

Residual Risk

With your inherent risks and their extent assessed, you can now evaluate the controls currently in place against each identified risk. This produces your residual risk assessment - the risk that remains after the application of existing controls.

Residual risk is calculated in the same way as inherent risk and the gap between the two is analytically important. A large gap suggests your controls are working effectively. A small gap - where residual risk sits close to inherent risk - indicates that your current controls are providing limited mitigation and that further action is likely warranted.

See table below for example:

Process Steps				Inherent Risk				Controls	Residual Risk		
OperationType	**Operation**	**Step No**	**Step Description**	**Risk Description**	**Impact**	**Likelihood**	**Score**	**Controls**	**Impact**	**Likelihood**	**Score**
Manual	Set up of Reinsurance Policy	**1**	The Policy Technician uses the information in the Underwriters Reinsurance Instructions to record the policy in the Reinsurance booking system.	The data may not be transcribed correctly.	High	High		A 100% QA check takes place over the set-up of the facultative cover in the booking system. This includes: 1. Checking the Underwriter Reinsurance Instructions to ensure that the facultative cover is linked to the correct direct policy and that the excess point is recorded corrrectly. 2. Checking that the inception and expiry dates on the facultative entry match those in respect of the direct policy on the inwards policy registration system.	Low	Low	
Manual	Inward Claim proceessing	2	The claims adjuster checks the details of the inwards claim against the policy to determine the claims validity. If the claim is deemed as valid, the adjuster initiates the payment process.	The inwards claim may not trigger the reinsurance recovery.	High	Medium		A monthly report is run to check whether any direct policies with facultative cover attached have had a paid claim but no reinsurance recovery initiated. The technician checks each entry and must document the reason for the lack of a recovery.			

Pro-tip: Organisations frequently find this assessment more difficult than it appears, not because the methodology is complex, but because objectively evaluating the effectiveness of your own controls is genuinely hard. There is a natural tendency to credit controls with more mitigation than they actually provide - particularly where controls are technically present but inconsistently operated, or where they were designed for a version of the process that has since evolved. Where possible, this assessment benefits from challenge and independent scrutiny.

Risk Appetite

With your residual risk picture established, you can now determine whether the remaining risk sits within your appetite. Where it does, no further action is required. Where it does not, you have a clear, evidence-based mandate to design additional or enhanced controls - and, critically, you have the risk context needed to design them well.

This is the point at which risk identification and data flow materiality assessment become directly actionable. The higher the inherent risk, and the weaker the existing mitigation, together with the higher materiality of the flow, the more investment in controls is justified. The framework you build in the next stage should be risk-proportionate - and the work you have done here is what makes that proportionality possible.

Designing Your Data Control Framework

Completing a thorough risk identification exercise is a significant achievement. You have mapped your data flows, understood your use cases and developed a clear picture of where your vulnerabilities lie. But this is the point at which many organisations make a critical mistake - they treat the risk identification as the destination rather than the starting point.

Without a properly designed Control Framework to act on those findings, what you have is visibility of risk, not mitigation of it. Visibility without action provides little protection when something goes wrong.

Why Most Frameworks Underperform

In our experience, Control Frameworks fail for predictable reasons. They are built reactively - controls layered on in response to incidents or audit findings rather than designed systematically against identified risks. They rely too heavily on a single type of control, typically detective, leaving the organisation exposed when that control is inconsistently operated or temporarily unavailable. Or they are technically sound on paper but impractical in operation, generating workarounds that quietly erode the assurance they were designed to provide.

The result is a framework that looks adequate from the outside but provides far weaker protection than the organisation believes it has. This is a particularly dangerous position to be in, because it creates false confidence at exactly the level - senior leadership, risk committees, regulators - where an accurate picture matters most.

Building a Cohesive System of Controls

The most effective frameworks don't rely on any single control. Instead, they deploy different types of controls across the data flow that work in concert, each compensating for the limitations of the others. When one control fails - and over time, they will - the others ensure that data quality is maintained.

Your framework should comprise two categories of control: **primary controls** and **compensating** controls.

- **Primary controls** are designed with a specific, identified risk in mind. They are targeted and precise, directly addressing the vulnerability you've uncovered in your data flow.

- **Compensating controls** provide a broader safety net. Their purpose is to maintain data quality even in the event that a primary control fails or is temporarily unavailable.

A formalised management review of summarised data prior to submission is a good example - it may not catch every individual error, but it provides a critical final check before data reaches its end use.

Within both categories, controls can be either **preventative** or **detective** in nature and your framework needs both.

Preventative Controls

Preventative controls are your first and most efficient line of defence. They are designed to stop poor data from entering your systems or processes in the first place, which means there is nothing to detect or correct downstream. System validations that restrict data entry to permissible values, or Master Data Management solutions that enforce consistency across systems, are strong examples of this type of control.

The efficiency of preventative controls makes them the natural foundation of your framework. However, they have limitations that are important to understand. A system validation can restrict what values are entered, but it cannot guarantee that the correct value has been chosen from those available. There is also the practical reality that system enhancements take time to implement, meaning that gaps will exist during transition periods. Preventative controls alone are therefore rarely sufficient.

Detective Controls

Detective controls complement your preventative layer by subjecting data to review and verification. Reconciliations to confirm that data has passed between systems completely and accurately, QA checks against a golden source and Data Quality Indicators all fall into this category.

These controls are important, but it's worth being clear-eyed about their nature: they are reactive. When a detective control identifies an issue, data has already entered your flow in a compromised state. Corrections will be needed and depending on how far downstream the issue has travelled, those corrections can be costly and disruptive. This is precisely why detective controls should complement a strong preventative layer, not substitute for it.

Practical Considerations

A Control Framework that looks robust on paper but is unworkable in practice will not be operated consistently, and an inconsistently operated control provides far weaker assurance than its design suggests.

When designing your framework, you need to weigh three factors carefully.

1. **Compatibility with your existing systems and processes**: controls that sit naturally within established workflows are far more likely to be operated reliably.
2. **Cost-effectiveness and feasibility:** every control carries a cost in time, resource, or both, and that cost must be proportionate to the risk being mitigated.
3. **Operational efficiency:** a framework that unduly slows down or impedes business processes will generate resistance and ultimately lead to workarounds.

This is where the criticality assessments you conducted over your data flows become directly actionable. The greater the criticality, the more investment in controls is justified. For lower-risk operations, a lighter-touch approach may be entirely appropriate. The framework should be risk-proportionate, not uniform.

Getting this balance right is not straightforward, but it is achievable. Organisations that invest the time to design their Control Framework properly - rather than layering controls reactively in response to incidents - consistently find that they spend less time and resource managing data quality issues over time, and more time trusting their data to do what it's supposed to do.

The Difference a Structured Approach Makes

Organisations that invest in designing their Control Framework properly - rather than building it incrementally in response to problems -consistently find themselves in a stronger position. They spend less time managing data quality issues reactively. They face internal audit and

regulatory scrutiny with greater confidence and they are able to place genuine trust in the data underpinning their most critical decisions.

Reaching that position is achievable. But it requires the right methodology, applied rigorously and with a clear understanding of what good looks like in practice.

Data Flow Mapping Checklist:

This checklist is designed to be used by the team conducting the data flow mapping exercise as a quality assurance tool, both during the mapping process and on completion. Its purpose is twofold:

1. To reduce the risk of gaps emerging through the exercise that will silently undermine your risk identification downstream

2. To provide a defensible record that your mapping was conducted rigorously.

It is not intended to replace the judgment of the people doing the work. Rather, it is a structured prompt to ensure that the most common sources of incompleteness - undocumented shadow processes, unreconciled cross-functional accounts, assumptions treated as facts - have been actively considered and addressed rather than overlooked under time pressure.

Where a point cannot be checked, that should be recorded as an open item with a clear owner and resolution date, rather than left unmarked. An incomplete checklist that is honest about its gaps is considerably more useful than a complete one that is not.

1. Define Your Use Cases First

- Have you identified all end-use cases that depend on this data (regulatory returns, management reports, models, etc.)?
- Have you used those requirements to prioritise which flows to map first and in most depth?

2. Business Process Understanding

- Have you engaged directly with the people doing the work, not just reviewed existing documentation?
- Have you documented the process as it is actually operated, not just the formal documented version?
- Have you specifically probed for shadow processes, manual workarounds, and undocumented interventions?
- Have you noted where human judgment is exercised within the process and what variability that introduces?
- Have you identified who owns each process step?

3. Technical Lineage

- Have you traced horizontal lineage - the movement of data from system to system across the full flow?

- Have you traced vertical lineage - all transformations, aggregations, recalculations and re-codings applied at each stage?
- Have you identified where data is combined with other data sources and documented those joins?
- Have you confirmed lineage with IT rather than relying solely on documentation?

4. Controls Documentation

- Have you documented all controls currently operating at each step in the flow?
- Have you recorded who is responsible for operating each control?
- Have you made an initial assessment of whether each control is consistently operated in practice?
- Have you flagged any controls that were designed for a prior version of the process and may no longer be fit for purpose?

5. Cross-Functional Input and Quality

- Have you obtained input from Business, IT, and Risk/Compliance functions?
- Have you reconciled any conflicting accounts between functions rather than defaulting to one view?
- Have you clearly distinguished between confirmed facts and assumptions in your documentation - and minimised the latter?
- Have you had the completed flow reviewed by someone with an independent perspective, ideally from outside the team being mapped?

6. Completeness Sense Check

- Does your flow run end-to-end from the original data source to the end use case?
- Would a new joiner be able to follow the flow without needing to ask for clarification?
- Are there any steps in the flow where you have lower confidence in the accuracy of your documentation?
- Have gaps or assumptions been explicitly flagged for follow-up rather than left undocumented?

Individual Control Design

You can invest significant time and resources in mapping your data flows, identifying your risks and designing a Control Framework and still find that it fails to deliver the assurance you need.

In our experience, this is where most organisations encounter the same problem: **the controls themselves are too loosely defined to be operated consistently or audited reliably.**

It is a more common issue than many risk and data teams realise and it tends to surface at the worst possible moment - during an internal audit, a regulatory review, or in the aftermath of a data quality incident that a well-operated control should have caught.

The Cost of Ambiguity

A control that lacks a precise, documented methodology will be operated differently by different people and differently again following staff turnover. Over time, what was intended as a robust detective or preventative measure quietly becomes a box-ticking exercise - present on the control register, but providing little genuine assurance.

This is not a reflection of poor intent on the part of the people operating the controls. It is almost always a design problem.

Design Effectiveness and the Need for Robust Control Narratives

For a control to be effective in design, its operation must be unambiguous and repeatable. Anyone responsible for operating the control - whether today or in twelve months' time following staff turnover - should be able to do so consistently and correctly based on the control narrative alone. Equally, any second or third line reviewer should be able to assess whether the control has been operated as intended without needing to seek clarification.

To achieve this, every control narrative must address five fundamental questions:

Who is responsible for operating the control? This should identify the specific role accountable for its execution, not simply a department. Ambiguity over ownership is one of the most common reasons controls fail in practice.

What is the nature of the control? Is it a reconciliation, a QA check, a system validation, a management review? The type of control should be clearly stated as it shapes everything that follows.

How is the control to be operated? This is the methodology - the step-by-step process that the responsible party follows each time the control is executed. The more precisely this is documented, the more consistently it will be applied.

Why does this control exist? Documenting the specific risk the control is designed to mitigate serves two purposes. It ensures the control remains fit for purpose as the business evolves and it provides context that helps operators understand the importance of rigorous execution rather than treating it as a box-ticking exercise.

When must the control be operated? Frequency is a design choice that should be driven directly by the risk. A control over data feeding a daily regulatory submission has very different timing requirements to one covering a monthly management report. Getting this wrong - operating a control too infrequently relative to the risk - is a common and consequential design flaw.

Finally, the narrative must specify what evidence is to be retained as a record of the control's operation. Without this, your second and third lines have no basis on which to assess operating effectiveness and you have no audit trail in the event of a challenge.

Putting It Into Practice

The following example illustrates what a well-constructed control narrative looks like in practice, using an operational QA check over policy and premium registration data in an insurance company:

The QA Analyst undertakes a daily QA check over the previous day's policy and premium registration bookings. The purpose of this control is to ensure the accuracy of the 30 identified Critical Data Elements.

The QA Analyst operates the following process:

1. *Run the Bookings Report for the previous working day's data*
2. *Select at random 40% of bookings, ensuring a proportionate split by team code*
3. *Compare the data elements between the booking system and the Front Sheet*
4. *Record any discrepancies and refer them to the relevant Technician for explanation or correction*
5. *Escalate any non-responses by the Technician to the QA Lead after two working days*

The outcome of each data element checked is recorded in the Workflow system, together with any referrals made to the Technician and the responses received.

What makes this narrative effective is its precision. The scope is defined, the methodology is replicable, the escalation path is clear and the evidential requirements are explicit. A member of Internal Audit picking this up for the first time has everything they need to assess whether the control has been operated correctly - and the person operating it has no room for ambiguity about what is expected of them.

This level of rigour should be applied consistently across every control in your framework. It is not an administrative burden - it is the mechanism by which your framework delivers the assurance it promises.

Recognising the Gap

If any of the following resonate, it is likely that your current control design warrants closer examination:

- Your controls are documented, but the narratives vary significantly in detail and quality.
- Your second or third line has raised questions about the consistency of control operation.
- You have experienced data quality issues that existing controls should theoretically have prevented.
- Key person dependency means that controls are operated correctly only when specific individuals are in post.

These are not edge cases. They are the norm in organisations that have built their Control Frameworks incrementally, without a structured design methodology behind them.

Is Your Data Controls Framework Truly Fit for Purpose?

Data quality risk is not a problem that resolves itself. Left unaddressed, it compounds - quietly undermining the decisions, returns and reports that depend on your data being accurate, complete and reliable. The organisations that manage it most effectively are not necessarily those with the most sophisticated systems.

They are the ones that have taken a structured, methodical approach to understanding where their risks lie and building a framework of controls designed specifically to mitigate them.

The stages covered in this guide - mapping your data flows, identifying inherent and residual risks and designing a Control Framework with the right balance of preventative, detective, primary and compensating controls - represent that structured approach. Taken together, they provide a clear path from uncertainty about your data quality risk to genuine, auditable assurance over your most critical data.

But reading about the methodology and applying it effectively are two different things. The gap between a Control Framework that looks adequate and one that genuinely performs under scrutiny is real and it is wider than many organisations realise until they are tested.

Where Do You Stand?

If you are unsure whether your current framework is built on sufficiently robust foundations, or if you recognise any of the challenges described in this guide within your own organisation, the right next step is a straightforward one.

A discovery call will give you an honest, independent assessment of where your framework currently stands, where the most material gaps are likely to lie and what a realistic path to remediation looks like. There is no obligation and no sales pitch -just clarity, from someone who has designed similar frameworks within the insurance industry as well as audited Chief Data functions.

The cost of that conversation is half an hour of your time. The cost of not having it tends to be considerably higher.

Book your discovery call here: https://calendly.com/navin-ahuja-provenancedatarisk/30min

Appendix: Framework Assurance Summary Checklist

This checklist is intended for senior stakeholders - risk committees, internal audit and oversight functions - as a high-level means of assessing whether the organisation's Data Control Framework has been built on sufficiently robust foundations. It is not a substitute for the detailed work documented elsewhere in this guide, but a structured prompt for the conversations that should be happening at leadership level about the quality of that work.

Data Flow Mapping

- ✓ Have all critical data flows been mapped end-to-end, from source to use case?
- ✓ Was the mapping exercise conducted with active input from Business, IT, and Risk/Compliance functions?
- ✓ Has the completed mapping been reviewed by an independent party and validated as accurate and complete?
- ✓ Is there documented evidence of the mapping exercise that would withstand external scrutiny?

Risk Identification

- ✓ Have inherent risks been identified at a sufficient level of precision to support meaningful control design?
- ✓ Have likelihood and impact assessments been made honestly, with evidence, rather than defaulting to conservative gradings?
- ✓ Has the effectiveness of existing controls been assessed objectively, with independent challenge where possible?
- ✓ Is there a clear picture of where residual risk sits outside appetite and what action is planned in response?

Control Framework Design

- ✓ Does the framework address the identified risks systematically, rather than having been built reactively in response to incidents or audit findings?
- ✓ Does it deploy both preventative and detective controls, with neither relied upon exclusively?
- ✓ Does every control have a documented narrative that is precise enough to be operated consistently and audited reliably?
- ✓ Is the framework proportionate - with control investment concentrated where inherent risk is highest?
- ✓ Has the framework been stress-tested against the question: would it perform under regulatory scrutiny or in the aftermath of a data quality incident?

We hope you found this guide useful.

Feel free to submit your suggestions for additions to the PROVENANCE METHOD here:

navin.ahuja@provenancedatarisk.com

Navin Ahuja | LinkedIn