More Than Just Code: Organizing for Change on a Modern Data Platform

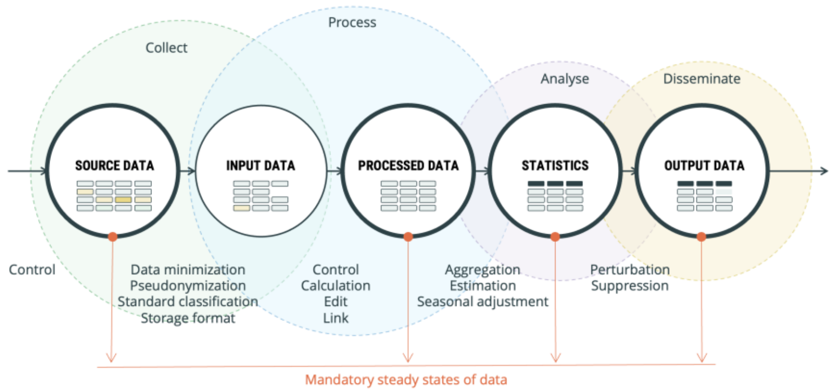
Author: Trygve Tatsuya Falch, Head of IT-Architecture (trf@ssb.no), Statistisk sentralbyrå (Statistics Norway)

Kommentoinut [JH1]: Add you e-mail address and position at the office

Abstract

Transitioning a national statistical institution to a cloud-based data architecture involves intense operational friction. The development of Dapla, the modern data platform at Statistics Norway, began with conflicting methodologies and systemic missteps. This paper outlines a process of iterative trial, error, and course correction. This approach depends on an executive leadership culture that accepted operational failures and facilitated rapid course corrections. To structure our practical experiences, we anchored our empirical lessons in W. Edwards Deming's systems thinking [1] and the DevOps findings of the Accelerate framework. These external theories provided the direction to recognize that delivery velocity and system stability reinforce each other within data production. Our eventual framework relies on Just-in-Time governance through the Plan-Do-Check-Act cycle, minimizing unmanaged shadow infrastructure. Reliance on open-source software and open programming languages establishes a collaborative framework across national boundaries. Organizations must also shift from task-oriented project structures to product-focused value streams to lower publication lead times. On a technical level, data production is decoupled through defined architectural checkpoints called Steady States. This model shifts the IT department's role to that of a provider of self-service platforms. Domain experts gain the autonomy to manage their statistical production independently, supported by systematic organizational training and targeted recruitment. The operational boundary between technology units and domain teams evolves alongside changing recruitment priorities and advancing organizational maturity. Modernization succeeds when institutional leadership removes operational obstacles, enabling statistical producers to actively direct their own technological tools.

Kommentoinut [JH2]: Mark this as abstract. Typically in abstracts there are no references.



1. Theoretical Foundations of Statistical Modernization

Statistical production architectures require structural modernization to accommodate new data sources and cloud-based environments. Modernization frameworks frequently overemphasize technical code while ignoring the organizational structures that support it. Management theorist and quality pioneer W. Edwards Deming, known for shaping modern industrial efficiency and Lean thinking — established that ninety-four percent of operational problems belong to the system itself rather than individual employees [1]. Management must focus on redesigning organizational processes. Demanding better performance from individuals fails to address systemic flaws. This systemic perspective aligns with contemporary software engineering insights. The Accelerate research program analyzed data across multiple years to identify the core drivers of organizational performance [2]. The findings demonstrate that deployment velocity and system stability are not competing trade-offs; they act as mutually reinforcing metrics. For a national statistical office, developing a capability for continuous structural change is an operational necessity. Technology shifts rapidly, requiring identical velocity in institutional adaptation. Newer software platforms and programming tools are simply modern mechanisms to deliver the same trustworthy statistics that society has relied on for generations.

2. The Messy Reality: Migration, Simplicity, and the Mandate to Fail

Implementing a modern data architecture introduces immediate operational friction. Statistics Norway encountered this complexity during the development of its data platform, Dapla. The institution had to maintain legacy production pipelines while constructing an unproven ecosystem. This process resembles paving a road while driving upon it. The organization had to redesign the vehicle and the institutional structures simultaneously. Friction occurs when a platform team attempts to solve urgent user demands before the underlying technical foundation stabilizes. This environment requires an adaptive operational mindset. The organization must treat the transition as an active structural transformation. Migration is not a static software delivery.

During this developmental phase, technical solutions grew overly complex to meet perceived or assumed requirements. The platform team established a core architectural principle to challenge established assumptions about the inherent complexity of data systems, prioritizing simplicity instead. While complexity inevitably manifests in different parts of an enterprise architecture, making the entire organization recognize that complexity carries a heavy institutional cost allows teams to collaborate. This cost extends far beyond the information technology department. When domain departments understand this shared burden, the organization can collectively design lighter, more agile, and more maintainable solutions.

The final architectural design was not apparent at the beginning of the project. The platform team evaluated multiple conflicting methodologies and encountered numerous systemic failures before establishing the current operational framework. Acknowledging this initial uncertainty is necessary for a realistic assessment of institutional modernization. This structural journey depended entirely on executive leadership that actively tolerated operational failure. Management provided the psychological and structural security required to execute rapid course corrections when errors occurred. Without an executive mandate that protects teams during experimental mistakes, technical staff default to risk-averse behaviors that stifle architectural evolution. The institutional allowance for failure, combined with active leadership backing, constitutes the primary driver of long-term platform stability.

Statistics Norway avoided recreating independent technical silos by leveraging open-source assets and collaborating with domestic and international peers. The department engaged with the Norwegian Labour and Welfare Administration and the French national statistical institute. Adopting open-source software and open programming languages, specifically R and Python, provides a common statistical language. Combining these open tools with cloud-native design creates a portable ecosystem. Statistical solutions can move across international borders without requiring extensive IT infrastructure intervention.

3. Just-in-Time Governance and the Autonomy Fallacy

Maintaining speed during an enterprise-wide platform migration requires a departure from traditional governance. Organizations often build restrictive security fences based on theoretical risk assumptions before real users open the system. This practice introduces unnecessary bureaucracy. Statistics Norway applied Deming's Plan-Do-Check-Act cycle to its administrative and governance frameworks [1]. The platform team plans minimal viable support structures, deploys them directly into production, checks where users encounter operational friction, and acts to formalize successful solutions into enterprise standards. Roads are paved precisely where user traffic naturally flows. Effective governance solves the practical problems of technical experts. Governance frameworks should not serve to soothe administrative anxiety. This responsive model ensures that security guardrails remain relevant so that operational velocity is sustained.

However, decentralized execution presents specific risks. Misinterpreting team autonomy as a total absence of technical standards results in institutional chaos. When left without structural alignment, independent teams reinvent identical technical components. Statistical experts exhaust their capacity by managing low-level technical infrastructure and complex data pipelines. This layout increases the cognitive load on delivery teams and diverts energy from data science and core statistical production [4]. Autonomy fails to benefit statisticians if it forces them to become infrastructure engineers. True operational autonomy relies on shared enterprise guardrails. Well-defined platform constraints remove technical noise so that specialized personnel can focus on data analysis.

4. The Constraints of Centralization and Generic Systems

Centralized information technology departments frequently favor generic, standardized applications because they reduce maintenance overhead. This preference creates an operational bottleneck known as the generic trap. Standardized software tools successfully resolve eighty percent of common enterprise tasks. They frequently block the final twenty percent of specialized execution. This final twenty percent represents the exact stage where the unique scientific value of official statistics is generated. Forcing highly specialized analysts into restrictive, generic software boxes does not stop their development work. Instead, users migrate their innovative work to unmanaged shadow systems outside the corporate platform. Platforms must support the unique requirements of the specialist. Optimizing exclusively for the average user creates operational friction.

Tightly coupled, centralized architectures generate substantial hidden liabilities. Modifications within a unified system create an unpredictable blast radius. Minor adjustments in one component trigger unexpected failures across unrelated pipelines. This architectural fragility creates a risk-averse institutional culture. Engineering capacity is consumed entirely by the maintenance of legacy debt and basic operational upkeep. Innovation ceases when every technical adjustment requires centralized coordination or dedicated IT resource allocation. Velocity requires the systematic decoupling of both the technical systems and the organizational units [4].

5. The Product Framework: Value Streams and Steady States

Generating speed requires a shift from a service-desk fulfillment model to an engineering empowerment model. The information technology department must build a stable, intuitive platform that allows statisticians to work without submitting service tickets or waiting for manual handoffs. Self-service infrastructure eliminates administrative delays. This shift transfers complete ownership of statistical production to domain experts. It gives operational authority back to the personnel who understand the underlying data.

Statisticians are not automatically fully competent data scientists when given this expanded autonomy. However, establishing these clear performance expectations and anchoring them within both domain leadership and executive management marks a fundamental shift in the relationship between IT and statistical departments. This cultural shift is supported by systematic schooling, the creation of internal professional communities, and deliberate, targeted recruitment.

This architecture necessitates a continuous evaluation of operational boundaries between the information technology department and the statistical production teams. The division of labor between the platform capabilities delivered by technology units and the responsibilities maintained by statistical teams remains dynamic. This operational boundary shifts as the institution alters its competence focus during recruitment and as the overall maturity of the organization develops. The division of labor requires ongoing calibration to balance core infrastructure provision with specialized domain execution. The institution must constantly identify and remove structural obstacles that prevent statistical personnel from participating in their own modernization. When infrastructure barriers are systematically dismantled, those responsible for statistics are empowered to actively contribute to and direct the technical evolution of their specific domains.

This model integrates Mik Kersten's product framework [3]. Projects have definitive endpoints. Data products require continuous lifecycle management. The organization must transition from completing isolated tasks to optimizing end-to-end flow. Eliminating traditional handoffs prevents knowledge loss and removes systemic wait times. The production team maintains comprehensive ownership from the initial data source to the finalized output. Eliminating these institutional handoffs addresses a critical metric within statistical production quality frameworks. It directly lowers the total lead time between initial data collection and official publication.

Operational independence across these value streams is maintained through architectural checkpoints called Steady States. The platform team does not police the internal methodologies or processes used by statistical teams. Instead, the platform validates the specific state of the data product at predetermined landing pads. These checkpoints span across source data, input data, processed data, statistics, and final output data. If a dataset satisfies the automated quality indicators, it is trusted by the system. The Steady State acts as a clear data contract that allows different parts of the organization to operate independently.

6. Cultivating Human Capability

A sustainable technical architecture requires an equivalent investment in human capability. Modernization involves scaling human skills alongside platform capabilities [4]. This layout requires a balanced professional compact between separate disciplines. The platform engineering team must cultivate deep domain empathy to build infrastructure that reflects the reality of statistical science. Simultaneously, statisticians must develop platform literacy to manage their workflows independently within the cloud environment. Training initiatives must focus on core competencies such as version control using Git, code sharing, and practical platform implementation. When technical and scientific personnel share a common operational language, structural wait times disappear. The objective of an infrastructure transition is the establishment of a continuous, institutional capability for change. Success is defined by long-term organizational agility rather than the completion of a temporary information technology project.

7. References

Kommentoinut [JH3]: Move references to next page.

- **[1]** Deming, W. E. (1982). *Out of the Crisis*. Massachusetts Institute of Technology. Available on Amazon: https://www.amazon.com/dp/0262541165
- **[2]** Forsgren, N., Humble, J., & Kim, G. (2018). *Accelerate: The Science of Lean Software and DevOps*. IT Revolution Press. Available on Amazon: https://www.amazon.com/dp/1942788339
- **[3]** Kersten, M. (2018). *Project to Product: How to Survive and Thrive in the Age of Digital Disruption*. IT Revolution Press. Available on Amazon: https://www.amazon.com/dp/1942788398
- **[4]** Skelton, M., & Pais, M. (2019). *Team Topologies: Organizing Business and Technology Teams for Fast Flow*. IT Revolution Press. Available on Amazon: https://www.amazon.com/dp/1942788819
- **[5]** The Standards Comittee (2026), *Steady States of Data in SSB*. https://www.ssb.no/en/teknologi-og-innovasjon/metoder-og-dokumentasjon/steady-states-of-data-in-ssb-version-3.0