European Research Studies Journal
Volume XXVIII, Issue 3, 2025

pp. 1518-1537

Project-Lifecycle Data as the Conceptual Basis for Data-Driven Governance in Global IT Programmes

Submitted 15/08/25, 1st revision 29/08/25, 2nd revision 11/09/25, accepted 30/09/25

Tomasz Zdybicki[1]

Abstract:

Purpose: *This paper presents a case study of a global IT programme in the pharmaceutical industry, demonstrating that a structured data model can serve as a basis of data-driven programme governance. Coherent, accountable, and predictable data flows play an important role in enhancing predictability, transparency, and operational efficiency of large-scale initiatives, the study contends.*
Design/Methodology/Approach: *Qualitative case study design was employed. As an active participant in the programme, the author did participatory observation, information-flow mapping, and data-model design as part of the research. Empirical material consisted of operational data from Jira, Monday.com, ServiceNow, and Google Sheets. Project documentation and meeting records also formed part of the evidence.*
Findings: *The analysis identified severe fragmentation of data sources as well as ambiguity of ownership, with implications for decision time delays, duplication of tasks, and less audit trail. The development yielded a consolidated model that was Single Source of Truth based. It helps to organize roles, processes, and responsibilities along common datasets in which responsibility for decisions is traceable, thereby providing a means of bringing back uniformity and responsibility for decisions made.*
Practical implications: *The proposed methodology can be utilised in another, equally complex programme to reduce coordination cost, build better data stewardship, and allow for evidence-based policy choice. It establishes an approach of logical integration of operational execution with strategic governance through orderly flows of information.*
Originality/Value: *By demonstrating that a data model constitutes a management approach, the research connects the project governance and data governance literature. It provides a repeatable model for converting decentralized project ecosystems into coherent, data-driven governance systems.*

Keywords: *Data governance, single source of truth, programme management; information integration; systems thinking.*

JEL codes: *M15, O32, D83.*

Paper type: *Case Study / Research Paper.*

[1]*University of Szczecin, Institute of Management, Poland, ORCID: 0009-0000-0350-8270, e-mail: tomasz.zdybicki@usz.edu.pl*

Acknowledgement: *This research was co-financed by the Minister of Science under the "Regional Excellence Initiative.*

Regionalna Inicjatywa Doskonałości

Ministerstwo Nauki i Szkolnictwa Wyższego

1. Introduction

This article describes the findings from a case study implementation in the pharmaceutical industry of an international IT program. The service provides online solutions to more than one hundred company affiliates globally.

Running such a complex program without reliable data and a single source of truth is like being a cognitive blind man. When all this occurs in the absence of objective information, projects lag, costs overrun, and progress becomes difficult to control.

People need to understand what's happening in various areas simultaneously, trust what's driving decisions, and be able to invest and use resources accordingly. The only way a program can remain in good standing is through organized planning of future investments in terms of operational capabilities (Büchner *et al.,* 2010).

Management is not only about "data," but about data that is reliable, consistent, and timely. Data must reflect the program coherently, without contradictions, duplication, and unauthorized parallel versions. Only then can it serve as a credible basis for decision-making, progress tracking, and risk analysis.

Equally vital is the existence of this data, metrics, for instance, of team performance (velocity). For any organization, these are essential tools for understanding delivery capacity and ensuring an accurate assessment of future investments.

The program analysed operates around seven major projects in parallel, sharing similar data sources and infrastructure. Every month, Software Production delivers new releases that must be coordinated with the plans for each initiative and the readiness of each country for deployment. Scrum and Kanban methodologies are used to structure teams in the program but don't provide the consistency of cross-project data at this scale.

Distributed teams, spread across countries and time zones, directly affect well over one hundred local affiliates. About 120 people are involved in the program, coinciding with the number of countries supported. With such complexity, people are forced to rely on improvisation: developing private spreadsheets, introducing informal communication mechanisms, and organizing meetings purely to confirm facts.

Thus, time tends to be wasted enforcing alignment rather than achieving delivery. But while far from ideal, these ad-hoc practices keep the program on track, thanks to a group of skilled "stars", people who can fight fires and nip problems in the bud. However, constant improvisation, endless meetings, and siloed communication require enormous time and energy just to reconcile information. The current operational model is too complex to continue and will eventually lead to system overload.

Therefore, the program sponsor set one big goal: to reduce unpredictability and the time taken to deliver by moving from a *Time & Material* model to a *Managed Service* model, which would be more predictable. The new approach aims to increase overall productivity and delivery repeatability by 30–50% without requiring more resources.

This means fundamentally reorganizing how project data and information flow up and down the pipeline, as well as decisions, from which information is drawn. Existing studies around IT project management mainly focus on processes, methodologies, and team communication, not the data organization itself, the very basis for predictability and performance.

This article addresses the aforementioned problem by showing that structuring and visualizing data flow can provide an efficient approach to addressing complexity in IT program management. The idea is to show that projects can be structured from data, with the means to organize itself, to define where the data begins and ends, and how those data are owned and dependent. After analysing the flow of information and the design of the program's data model, the author suggests a framework to handle large-scale projects with fewer resources and fewer coordination meetings between different stakeholders.

2. Literature Review

While an increasing number of IT project management literatures focuses on how data is applied, it is becoming clearer that data is a key component in decision making and the continuity of organisational agility. Data quality, integration and consistency are critical for successful project delivery and, consequently, an organisation's ability to respond to change (Brous *et al.,* 2020; Naguib *et al.,* 2024).

The absence of integrated data sources results in information asymmetries, decisions in error and prolonged implementation timelines, particularly during complex project settings (Whyte and Davies, 2021).

Data governance is to guide and oversee data processes and the collection of data information including information creation and data consumption, from inception through to end-consumer management, so that that information is accurate, consistent and secure (Acev *et al.,* 2025; Grima *et al.,* 2023; Tyagi *et al.,* 2023).

Good data governance necessitates not only technically driven structures, but critically roles and responsibilities, and common standards of validation (Janssen *et al.,* 2020). In this light, data should not only be an asset for operations but also a managerial tool that shapes decision making across programmes.

Newer approaches like model-based governance and model-driven decision-making view data, processes and roles as elements of an integrated (Armenia, 2019). A system-thinking approach sees the organisation as a network of related items whose changes in data or decision processes induce operational and systemic effects (Barnabé and Perissi, 2019).

Research on large, international projects shows that distributed data environments decrease predictability, lead to greater coordination meetings, and frequently lead to ad hoc decision-making (Aramali *et al.,* 2024; Whyte and Davies, 2021). Without data integration, there is a risk for regulatory non-compliance as well as further reporting issues and a lack of objectivity and consistency in the organisation's measurement tracking progress.

On the other hand, research on data governance (Acev *et al.,* 2025; Jiya, 2021) emphasizes that high quality data, managed under clear ownership and stewardship – is crucial to establishing trust and transparency in governance processes. The perspective however, is rare, as it falls well out of scope with data being a significant component in IT project management where data is used not only to provide information but also at a macro level of decision-making.

This article addresses this gap by suggesting a framework whereby the programme's data model evolves to be a significant governance component in its own right, becoming a Single Source of Truth to add predictability, transparency and efficiency to challenging IT-related programmes.

3. Research Methodology

3.1 Research Purpose and Approach

The objective of this study was to improve the performance of a global IT program by streamlining data flows and increasing the role of information in decision-making. In practice, this requires:

- defining the locations and functions where data is generated,
- reducing informational chaos through the introduction of a Single Source of Truth,

- defining ownership of individual data sets and their use in day-to-day operating processes.

A key part of the analysis was the need for data completeness to allow for data-related management of the programme (informed by reliable reports and management dashboards (BI dashboards)) to enable continuous monitoring of progress, efficiency (through performance indicators) and risk.

The research was exploratory and qualitative. The emphasis was on the observation of roles, processes, and flows of information in a project environment. The author, who was an active participant in the programme, undertook participatory observation and in a systematic way recorded phenomena occurring in normal management.

The approach adopted derives from the author's own professional experience in dozens of other projects – both within the pharmaceutical and automotive sectors – having faced similar challenges in data and information management. Building upon over 30 years of expertise in global IT projects, including in telecommunications and government domains, the author utilized his background in databases, information modelling, and business intelligence (BI) to develop a data-driven, pragmatic method of analysis.

3.2 Program Context and Data Sources

Research was conducted in a global IT programme of the pharmaceutical industry covering more than one hundred subsidiaries of a single organisation worldwide. Approximately 120 people were involved in software development, deployment, and maintenance. The programme covered seven projects running in parallel, with data, work resources, and schedules shared across each project.

The teams adhered to Scrum and Kanban principles but operated in environments with different time zones that required constant cooperation. Data from operations included Jira, Google Sheets, Smartsheet, Monday.com, ServiceNow, Excel, and internal project registers. Supplementary documents were minutes of meetings, rollout plans, team schedules, and product documentation. The information used covered both areas of planning (e.g., release management, capacity planning) and operations (e.g., progress, delivery scope, team efficiency).

3.3 Stages of Analysis

The current state analysis was performed in five phases:

1. Defining the roles and responsibilities – highlighting the role key stakeholders in a programme play, what is expected of them, and responsibilities for the generation of data.
2. Source and type mapping – data sets that the project produced are mapped

here too (e.g., rollout plans, backlogs, access management).

3. Analysis of meetings and communication systems – watching consistent meetings and formal channels to see what is generated, communicated, or approved.
4. Mapping information flows – visually looking at the dependencies between data sources, roles, and processes.
5. Data model development – constructing models and modelling the relationships between the primary entities that correspond to projects, requirements, and work items.

3.4 Methods and Tools Applied

The analysis combined three different methods, namely, participatory observation, system analysis, and process mapping. Taking an active part in the programme gave the author insight into how data was produced – not just through tools but also in conversations, decisions, and the day-to-day cooperation of the team.

Relationships among programme components were generated by data modelling and information flow mapping techniques that facilitated the extraction of not just the technical aspects of the structure, but the sense making of information flow. Not to develop a ‘full scope’ of a technical model but instead to understand how data “lives” in the organisation – who creates it, who it goes around, and how does it get transferred inside the different parts of the organisation and when is it indeed the place of decision-making.

Outside of document evaluation and observation, the author engaged in several informal conversations with members of the project team and users of the system. And it was often through these conversations that these micro-interactions began to show up in subtle ways, as people repeatedly asked the same questions, sought verification of what they knew, and referred to “that one person who knows for sure” multiple times.

Data gained through these observations allowed for a deeper comprehension of the programme’s needs for information and the extent to which existing data supported management – and under what conditions it merely gave a sense of control.

3.5 Research Limitations

This was a qualitative study of a single global programme, however, and was not developed in a vacuum. The practical approaches and lessons are also informed by the author’s knowledge of this work, gleaned from dozens of other projects – especially in the automotive industry – which all faced the same problem of fragmented data, organisational complexity, and constant cross-team synchronisation.

The analysis covered the entire project life cycle, including resource planning, scope definition, execution, and deployment. Although conducted within a specific environment, the approach that has been employed is a universal one: data and processes may be dispersed, but efficiency is achieved only through structure and governance.

Successful governance frameworks have well-defined architectures and well-structured processes (Acev et al., 2025). In reality, the organisation of data tends to serve as the foundation for structuring the whole programme, whatever its sector or size. The derived conclusions serve as the basis for building the programme's data model, which details the roles, processes, and information flows within the studied environment. The following chapter presents how these findings were translated into a coherent and operational data-governance framework.

3.6 Diagnosis of the Programme's Initial State

The organisation of the programme was defined by substantial complexity and fragmentation, both in team structures and information management. The formal protocols, such as the Agile Release Train (ART) framework, were applied, but there was weak central coordination between methodologies, procedures, and standards and overall efficacy was poor. Accordingly, teams functioned in parallel, with considerable independence, while the overarching governance mechanisms remained limited or purely nominal.

Roles and responsibilities were widely distributed across multiple teams. The same individuals often carried out different duties simultaneously, analysis, implementation, and quality control, which made it very difficult to keep clear priorities and hold each other accountable.

Project Managers (PMs) were part of the governance system but in practice spent most of their time coordinating day-to-day business activities and resolving existing issues rather than developing an integrated management and data framework. There was no Data Owner or Data Steward; data did not belong to any specific people and was updated by multiple unsupervised individuals. Therefore, the quality and completeness of information depended heavily on the practices of particular teams.

3.6.1 Data Sources and Lack of Integration

At the start of the analysis, programme-related data were scattered across multiple tools and Google Sheets. There was no centralised database or standardised model for collecting and updating information. Each team maintained its own working summaries that differed in scope, format, and update frequency. The following areas illustrate the diversity of data sources and solutions:

- Business priorities – recorded in Google Sheets and manually updated by business representatives.

- Release management plans – partly kept in Google Sheets (deadlines, scope) and partly in Monday.com (operational schedules and status tracking).
- Availability and absence planning – maintained in a separate internal Google Sheet, disconnected from release and resource planning.
- Team structure and roles – no central register of people, roles, experience level, or workload allocation.
- Sprint planning (production) – managed in Google Sheets, then manually transferred to Jira after each sprint.
- Deployment plans – stored mainly in Google Sheets, with local versions maintained independently by country teams.
- Other projects – tracked variably in Smartsheet, Google Sheets, or Monday.com.

The systems did not support the calculation of velocity metrics or generation of burndown charts. Without a central repository, each spreadsheet and table functioned independently, making it impossible to verify data integrity or synchronisation between tools. As a result, multiple versions of the same information (deadlines, scope, status, etc.) existed simultaneously – and none was recognised as authoritative.

As Whyte and Davies (Whyte and Davies, 2021) point out, systems integration is one of the core processes in project organisation, determining how deliverables are assembled and aligned within project execution.

3.6.2 Organisational and Behavioural Consequences

The fragmentation of data sources was so extreme that it also impacted team dynamics and the working environment. Because team members lacked confidence in the quality and speed of the data, they received enough to use this data for confident decisions, members of the team continuously had to check information to verify that they were making decisions with real-time accuracy. This was echoed all the way down – from business analysts and developers and testers to the management team (including project managers, product owners and technical leads).

As a result:

- the number of control meetings rose,
- communication between teams began to be reactive and mostly ad hoc,
- lots of people began making private copies of files just in case, which contributed to growing data fragmentation problems.

The organizational chaos, with its minimal formal order, meant that just about every small and consequential choice had to be made by whoever happened to agree at that time, often without an official record. Nobody, in many instances, knew who had made the decision and when that decision to change scope or time had been made.

In addition, with no clearly defined model of data and no definite assignment of responsibilities, it is difficult to pinpoint how decisions were actually made or when those were made. Even important information, like release dates or deployment plans, was not centrally available.

Accordingly, it was extremely challenging to assess the actual programme state, and it was impossible to measure team velocity and performance. Such strategic and operational decisions had to occur with little confidence in data reliability, severely constraining managerial capacity.

Such a high level of data fragmentation for the pharmaceutical sector also leads to the potential risks of regulatory non-compliance and loss of auditability in project decisions. Similar findings were observed by Janssen et al. (2020), who suggest that the lack of a clearly defined data stewardship and shared ownership approach erodes information trust and diminishes the capacity to make auditable decisions across complex environments.

3.7 Problem Synthesis and the Need for Change

The results of an analysis of the programme state at the outset indicated that the major issue was not individual operational issues (the main ones in the analysis did not originate from the individual person or the operation) but the problem of no central information control system or clearly defined data responsibilities. Unclear rules for how to update and responsibilities being spread out had teams functioning on their own realities.

Data, by contrast, did not offer one cohesive management tool, it was largely utilised as an operational tool for short-term tactical purposes. The current pattern of communication relied on improvisation and ad hoc exchanges, a temporary solution that did not produce predictable or reproducible work. Numerous meetings and informal exchanges were the norm in dealing with this, increasing organizational costs and cutting efficiency considerably.

As a result, critical decisions were taken in uncertain conditions and analysis of programme performances was limited to partial observation alone. The central issue was not that agile methods were unsuitable for the scale of the programme size, but that there was no overarching layer of data governance in place to provide uniformity and robust accountability all the way through a data lifecycle.

As many projects and countries are involved in an individual programme, a structured model of data writing and ownership and process was critical. The lesson from that analysis was that the programme needed good quality data, good, reliable data, one source of truth. Janssen et al. (2020) argue that transparent data ownership structures are fundamental for establishing credibility and trust in information, and as a result, for effective programme management.

There is a need for an open and transparent data accountability framework, supplemented by management dashboards and reporting tools that can assist progress monitoring and drive data-driven decisions. This prompted the creation of a programme data model and approach, a structure for defining specific roles, processes and data ownership to address the identified issues.

4. Target Data Governance Model

4.1 Logical Architecture of the Programme Data Model

To address the identified challenges, a programme data model was created to streamline information flows, assign ownership to specific data sets, and establish a Single Source of Truth. Not only does this model establish data structures, it also organises roles, processes and responsibilities throughout requirements' entire lifecycle, from business decisions to implementation and reporting.

The objectives, logical structure, thematic modules and the way it supports management processes within the programme are given in the next subsections. According to Armenia (2019), "Model-based governance integrates analytics and systems thinking to improve decision quality."

4.1.1 Objectives and Assumptions of the Data Model

Unified data structure:

The model shared platform was established as a common, coherent information structure which guides the programme for action. The aim is to reduce fragmented data, coordinate collection and updates of data and reporting between the analysis, planning and execution. The model captures the lifecycle of the requirements, from defining business needs through implementation and results reporting, ultimately acting as a foundation for programme performance analysis.

Single Source of Truth:

Single Source of Truth. Its premise is based on the principle that there should be one reliable data repository from which all programme roles can rely. Without local copies or manual spreadsheets, data quality and comparison across projects is guaranteed. As Hijazi *et al.* (2022) make plain, "A Single Source of Truth improves coordination and reduces duplication".

Roles and data ownership:

Each dataset is role-owned, updated through an agreed process and maintained at source for trust and sharing reasons in terms of data reporting.

Processes and role alignment:

Data model as a framework for running the fundamental programme processes from business analysis to software deployment. Each role is responsible for whatever data it makes, produces, or updates. Business Analysts define the requirements and connect them to Epics; Product Owners coordinate the releases and priorities; Project Managers drive schedules and inter-project dependencies; Release Managers manage release cycles and checklists. It is a common model that provides everyone with a single language and a single context in which data can be synthesized, maintained and reported.

Role of section 4.3 in the article structure:
This section does not give a technical review but shows how the data structure organizes roles and processes to hold people accountable for information. The model serves to effectively enforce a management discipline as more than a tool but as a governance system where decisions are made based on consistent, timely and trustworthy data.

4.1.2 Logical Structure of the Data Model

The data model captures a hierarchy of decision-making and responsibility throughout the lifecycle of requirements. Its design is well organized based on three tiers of management, strategic, tactical, and operational, yielding a steady stream of data from strategic and tactical decision-making to real-time technical implementation and results reporting.

The strategic part (management) includes elements to identify the route and priorities of the programme. A Business Requirement is a formal request issued by the Leadership Team or CAB during the Epic backlog review stages. For each requirement that is approved, it is assigned a unique identifier, which forms the basis for further analysis and planning.

Strategic decisions and the execution process in question are then reflected on the tactical level (planning/scope). Epics are made through this phase to group similarly functional or technical requirements that are tightly related. Delivery Units are planning tools assigned to individual projects. The Business Analyst, Product Owner and Project Manager jointly define, estimate and plan these elements.

The operational level (execution) covers all delivery-related activities and tracking of progress. It contains Work Items, specific units of work completed within a sprint and connected to Epics and Delivery Units. This layer incorporates relational tables like Release, Release to Environment, Entity, and Employee for deployment planning, progress tracking, and resource management as well.

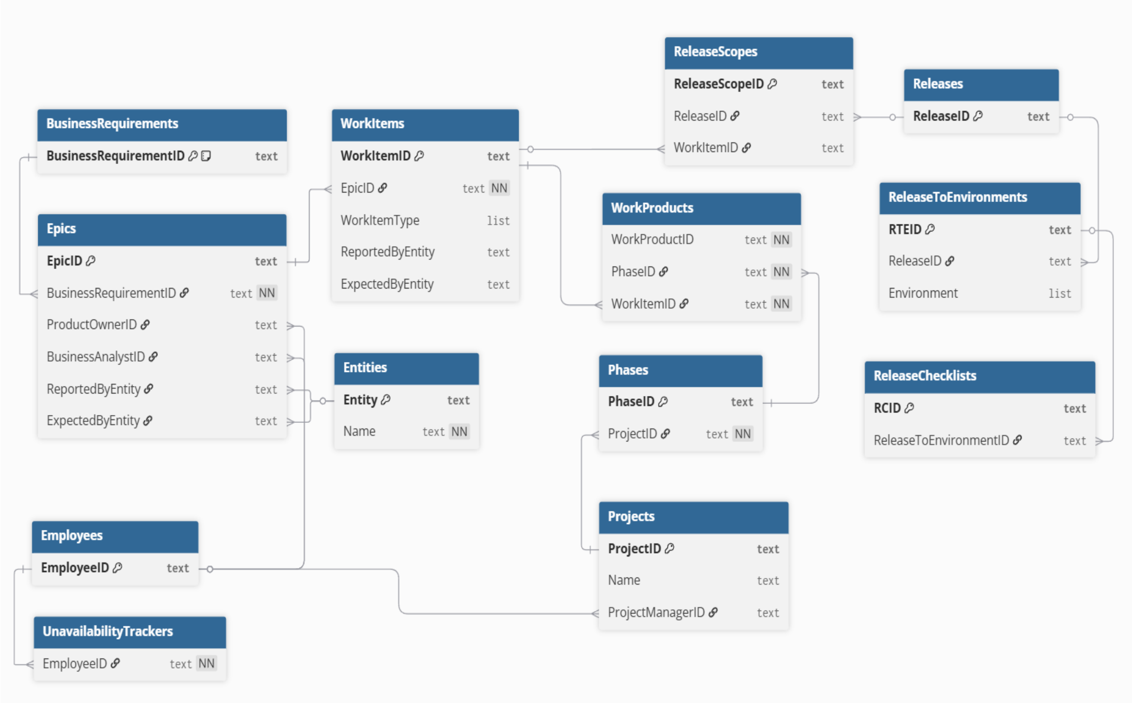
Each of these entities' relationships are presented in Figure 1 where we will see the logical setup of the programme’s data model from business requirements to their execution and deployment.

4.2 Programme Data Model – Entity Relationship Overview

4.2.1 Roles and Responsibilities within the Model

The following section presents a selection of key roles that play a critical part in the governance of the data model and ownership of its entities. These positions are not exhaustive but represent those with a specific influence on how information is structured, maintained, and utilised across the programme. The overview focuses on these roles from a process-oriented perspective, highlighting their contribution to data flow, decision-making, and operational alignment within the programme.

Figure 1. *Logical structure of the programme's data model; relationships between strategic, tactical, and operational entities.*



Source: *Own elaboration.*

Solution Owner – Represents the business; defines and communicates the product vision; co-develops the roadmap.

- IT Project Manager – Ensures alignment between business vision and delivery; tracks progress and KPIs.
- IT Lead – Ensures that the system design meets business requirements; defines IT strategy; supervises compliance and solution quality.
- IT Product Owner – Translates business vision into actionable plans; manages backlog and sprint goals; ensures transparency, clarity, and roadmap alignment.
- Program Manager – Oversees programme execution; manages risks, dependencies, and resources; ensures delivery of measurable outcomes.
- Business Product Manager – Defines product requirements; aligns the

roadmap with strategic business goals; bridges communication between business and IT.

- PMO Lead – Maintains programme standards and reporting; ensures data consistency; drives process improvement and delivery efficiency.
- Project Manager – Oversees scope, schedule, and risks; ensures timely delivery of planned outcomes.
- Release Manager – Coordinates release planning and deployment; facilitates Go/No-Go meetings; tracks release scope, progress, and outcomes.
- Functional Team Lead - Oversees the work of delivery teams, ensuring compliance with processes and data governance rules within their domain.
- Business Analyst (BA) – Gathers and analyses requirements; maintains traceability; supports testing and deployments across all levels of complexity.

4.2.2 Project Roles in Relation to the Data Model

Business Requirements:

Represent the strategic layer of business demand and define the initial scope of programme initiatives.
Owned by: Solution Owner.
Supported by: IT Project Manager, IT Lead, and Business Product Managers, who contribute to defining priorities, validating feasibility, and aligning requirements with the overall programme roadmap.

Epics:

Group related business requirements and define the functional or technical scope of delivery.
Owned by: IT Product Owners.
Supported by: Business Product Managers and Business Analysts, who ensure that epics reflect business priorities and are clearly linked to strategic requirements.

Work Items:

Represent the actionable scope of work for delivery teams, including acceptance criteria and linkage to specific releases.
Owned by: IT Product Owners.
Supported by: Business Analysts, who define, refine, and maintain the content and status of each item throughout the delivery cycle.

Projects and Phases:

Define the schedule and delivery plan for implementing Work Items over time.
Owned by: Project Managers.
Supported by: IT Product Owners, Business Analysts, and other project stakeholders who contribute to planning, tracking progress, and ensuring alignment with programme goals.

<u>*Releases and related entities:*</u>
Store release plans and their status, including dates, scope, and evidence of required activities.
Owned by: Release Manager.
Supported by: Functional Leads from development, deployment, and QA, who ensure readiness, quality, and compliance throughout the release process.

<u>*Employees and Unavailability Tracker:*</u>
Store information about staff availability, essential for assessing delivery capability and resource needs.
Owned by: PMO Lead.
Supported by: Functional Leads, who ensure that the data remains accurate and up to date within their respective teams.

This section only shows key aspects of the data model, but in reality, every project role operates on the model, usually unaware of their operation. We can imagine Scrum Masters estimating sprint capacity in terms of availability data held in the system, and user access to tools, environments, and communication channels lies in the Employees entity where it provides the user with individual roles and permissions.

Onboarding and offboarding processes, in similar fashion, directly work with employee data, every new team member means new assignments, new capacity adjustments, new access rights, and their departure sparks corresponding updates. Even developers and QA engineers are now using the daily status changes of their teams to continually update data which are then fed into reports from within the programme and dashboards, meaning the data model remains a live and inbuilt component of programme delivery.

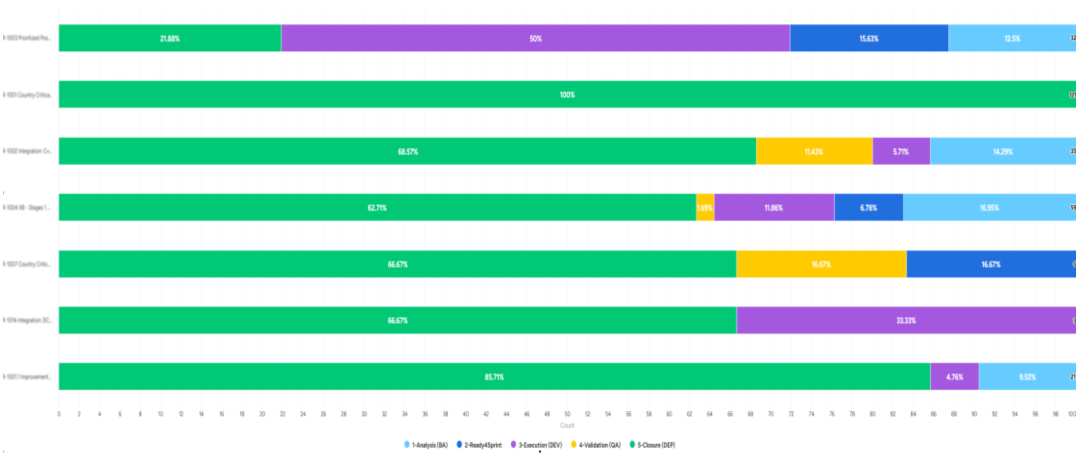
4.3 Validation of the Model through Operational Reporting

The present proposed data model has been constantly validated in the practical operation of the programme. Its architecture has been used on the production side for managing business requirements, release planning, and team capacity. This validation was carried out mainly in terms of operational reporting on live programme data which acted as a practical platform, confirming the model's consistency and applicability.

Alignment of entities, completeness of information, and the ability to monitor progress and dependencies in real time were clearly evident in reports. This was validated to show that the model demonstrates data consistency, reduces duplication, and facilitates traceability across multiple projects and countries. Observed improvements include faster reporting cycles, reduced coordination overhead, and higher trust in shared data.

To demonstrate the practical functionality of the proposed data model, representative report were generated directly from the programme's live data environment. The report serves as evidence that the model not only structures information effectively but also delivers reliable analytical outputs used in daily decision-making.

Figure 2. *Business Requirement Implementation Plan for the End of 2025.*



Source: *Own elaboration.*

Figure 2 depicts the progress of executing business requirements according to the status of related Work Items. The colour scheme shows the items' distribution across the various phases of the delivery cycle: Analysis, Ready for Sprint, Execution, Validation, and Closure. The report allows for immediate evaluation of the programme's overall progress and suggests areas in need of extra support or acceleration.

Figure 3 presents a view generated by combining information from *Business Requirements*, *Epics*, and *Work Items* (collected in Jira). It allows Product Owners and Business Analysts to quickly identify which Work Items remain to be developed in the context of requirement priorities.

4.3 Discussion: The Role of Reporting in Programme Management

It should be highlighted that the availability of reliable data sources not only underpins the operational management of projects in the teams' practice, but also the dissemination of operational knowledge within the organisation. For instance, if every information in the status field is used uniformly, every team member immediately knows what should be done next and so much less uncertainty or time delays result.

On the management end, tracking data across your programme enables programme leaders to follow up progress against plans, a value that is several times more useful than day-to-day task monitoring, as it enables you to ensure that collectively you are working towards the company's goals and that they meet the overall business and funding requirements.

Figure 3. *List of Work Items to be developed by Business Analysts.*

Source: *Own elaboration.*

Therefore, reporting is thus a key mechanism of programme governance: it changes raw data into situational awareness, informs decision-making, and provides accountability in distributed groups. Transparent data flows provide traceability to actions taken and enhance the organisation's ability to learn and adapt. As Brous (Brous et al., 2020), data governance is an essential success factor in relation to data science and analytics; Whyte and Davies (Whyte and Davies, 2021) also showed the absence of system integration increases coordination overhead and creates project predictability issues.

In a similar vein, Armenia (Armenia, 2019) highlighted that model-based governance facilitates transparency in decision-making as well as organisational learning. The model in this study operationalises these principles by indicating at least one direct effect on programme efficiency and coherence: in it, structured data ownership and a Single Source of Truth are demonstrated. As Few (Few, 2013) highlights, well-designed dashboards and data-visualisation practices are essential for converting information into actionable insight.

5. Conclusions, Proposals, and Recommendations

Given the high cost and complexity of large-scale programmes, they must be considered as systems operating through a data model. The precision with which this model is designed, its entities, relationships, and ownership structures, directly determines operational efficiency and decision quality.

Therefore, a dedicated role responsible for maintaining the integrity of the data model and data ownership should be established within the programme organisation. The cost of such a role is minimal compared with the potential financial losses caused by delays, miscommunication, or poor decisions.

Preliminary analysis indicates that adopting a structured data-governance methodology could lead to cost reductions of approximately 20-30% compared to the execution of an unstructured programme of a similar scale (around 100 participants). However, the full financial impact of this approach will become measurable after at least twelve months of continuous application.

A unique position that emerges within this model is the PMO Lead, whose responsibilities extend beyond traditional project-management governance. The PMO Lead not only maintains standards, schedules, and reporting frameworks but also acts as the custodian of the data model, overseeing relationships between entities, ensuring consistent data ownership, and enabling reliable analytical outputs and management reports.

In practice, the PMO Lead bridges the operational and analytical dimensions of programme governance, ensuring that project data remains credible and actionable so that the data model becomes the true backbone of decision-making and transparency.

5.1 System Fragmentation and Infrastructure Instability

An additional barrier to effective programme management through data and data models lies not only in limited awareness or competence but also in the instability of organisational infrastructure. Many companies regularly replace systems and vendors responsible for supporting project execution. Each new tool introduces another learning curve, and before teams can adapt, management decisions often trigger yet another round of replacements.

As a result, organisations operate multiple, partially overlapping systems that look impressive during demonstrations but lack proper interoperability or standardised data-exchange mechanisms.

Similar challenges with fragmented infrastructures and system interoperability have been discussed by Addo-Tenkorang and Helo (Addo-Tenkorang and Helo, 2016) who emphasise that frequent technology turnover and unaligned data architectures significantly reduce operational efficiency and long-term value creation.

This continuous technological rotation erodes efficiency, disrupts knowledge continuity, and prevents the establishment of a stable, integrated data environment necessary for consistent governance.

5.2 Security, Bureaucracy, and Cultural Constraints

Another challenge arises from corporate security policies and rigid IT governance. In an era of data-protection regulations and cybersecurity threats, many IT departments adopt a fortress mentality, protecting systems not only from external risks but inadvertently also from their own employees. Consequently, innovative internal initiatives often fail due to restricted access, lack of tools, or approval processes that can take months or even years.

Frequent system changes, limited technical support, and over-formalised procedures create an environment hostile to experimentation and continuous improvement. This paradox significantly limits the ability of programmes to leverage data effectively, even when competent teams and sound governance models are in place.

5.3 Organisational Responsibility for Data-Driven Governance

Effective data-driven governance requires clearly defined organisational ownership (Khatri and Brown, 2010). In the analysed programme, this responsibility is assigned to the PMO Lead and the PMO team, who act as custodians of the data model and its operational integrity. Their work goes beyond administrative coordination, they maintain consistency of data relationships, monitor data quality, and ensure that reporting supports evidence-based decision-making across the programme (Jiya, 2021).

The PMO team bridges the analytical and operational layers of the programme, maintaining alignment between data, processes, and delivery outcomes. By continuously monitoring data quality and supporting users in its correct application, the PMO acts as a practical enabler of data governance. This demonstrates that in large and distributed programmes, data-driven management must be anchored in a dedicated organisational role, supported by cross-functional collaboration and clear accountability structures.

5.4 Final Reflections

These observations suggest that sustainable data-driven governance depends not only on solid data structures but also on organisational stability and trust in the IT ecosystem. Long-term success requires more than tools or methodologies, it relies on the organisation's ability to maintain continuity, foster collaboration, and empower people to use and improve data systems freely.

It sparks a wider reflection on how organisations view the discipline of project and programme management. The project manager's role is often treated with a nonchalant degree in many IT environments, as someone who can "make things move," organize people, or briefly make discipline happen, all the while the technical and analytical skills are subordinate. But project management needs to be as careful as corporate finance management.

No organization would let its chief accountant work without clearly defined processes, data structures, and reporting standards. Still, improvisation is common in project management. With the volume of data produced by complex, multilayered projects comes the necessity to think of project management as a system: a defined data model with clearly defined roles, stakeholders, and data flows for real-time, on-demand visibility of the programme state.

It reinforces that data governance and programme governance must converge, data-informed decision-making should be based on both structure and evidence as opposed to intuition and personality. Only by aligning governance structures with cultural and infrastructural readiness can programmes fully realise the benefits of structured, model-based management.

Author Disclaimer: *The case study described in this paper is based on a real large-scale IT programme in the pharmaceutical. All examples and references have been anonymised for academic purposes. No confidential or proprietary data have been disclosed, and the opinions expressed are solely those of the author and do not represent the views of any organisation.*

References:

Acev, D., Biyani, S., Rieder, F., Aldenhoff, T.T., Blazevic, M., Riehle, D.M., Wimmer, M.A. 2025. Systematic analysis of data governance frameworks and their relevance to data trusts. Manag. Rev. Q. https://doi.org/10.1007/s11301-025-00545-1.

Addo-Tenkorang, R., Helo, P.T. 2016. Big data applications in operations/supply-chain management: A literature review. Comput. Ind. Eng., 101, 528-543. https://doi.org/10.1016/j.cie.2016.09.023.

Aramali, V., Gibson, G.E., Sanboskani, H., El Asmar, M. 2024. Enhancing project success: the impact of sociotechnical integration on project and program management using earned value management systems. Int. J. Manag. Proj. Bus., 17, 1-21. https://doi.org/10.1108/IJMPB-07-2023-0160.

Armenia, S. 2019. Smart model-based governance: Taking decision making to the next level by integrating data analytics with systems thinking and system dynamics, in: New Challenges in Corporate Governance: Theory and Practice. Presented at the New challenges in corporate governance: Theory and practice, Virtus Interpress, pp. 41-42. https://doi.org/10.22495/ncpr_10.

Barnabé, F., Perissi, I. 2019. Guest editorial. Kybernetes 48, 2-6. https://doi.org/10.1108/K-01-2019-548.

Brous, P., Janssen, M., Krans, R. 2020. Data Governance as Success Factor for Data Science. In: Hattingh, M., Matthee, M., Smuts, H., Pappas, I., Dwivedi, Y.K., Mäntymäki, M. (Eds.), Responsible Design, Implementation and Use of Information and Communication Technology, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 431-442. https://doi.org/10.1007/978-3-030-44999-5_36.

Few, S. 2013. Information dashboard design: displaying data for at-a-glance monitoring,

Second edition. ed. Analytics Press, Burlingame, California.
Grima, S., Thalassinos, E., Cristea, M., Kadłubek, M., Maditinos, D., Peiseniece, L. (Eds.). 2023. Digital transformation, strategic resilience, cyber security and risk management. Emerald Publishing Limited.
Janssen, M., Brous, P., Estevez, E., Barbosa, L.S., Janowski, T. 2020. Data governance: Organizing data for trustworthy Artificial Intelligence. Gov. Inf. Q. 37, 101493. https://doi.org/10.1016/j.giq.2020.101493.
Jiya, T. 2021. Responsible Data Governance in Projects: Applying a Responsible Research and Innovation (RRI) Framework. J. Technol. Manag. Innov., 16, 31-37. https://doi.org/10.4067/S0718-27242021000100031.
Khatri, V., Brown, C.V. 2010. Designing data governance. Commun. ACM 53, 148-152. https://doi.org/10.1145/1629175.1629210.
Naguib, H.M., Kassem, H.M., Naem, A.E.H.M.A. 2024. The impact of IT governance and data governance on financial and non-financial performance. Future Bus. J., 10, 15. https://doi.org/10.1186/s43093-024-00300-0.
Tyagi, P., Grima, S., Sood, K., Balamurugan, B., Özen, E., Thalassinos, E. (Eds.). 2023. Smart analytics, artificial intelligence and sustainable performance management in a global digitalised economy. Emerald Publishing Limited.
Whyte, J., Davies, A., 2021. Reframing Systems Integration: A Process Perspective on Projects. Proj. Manag. J. 52, 237-249. https://doi.org/10.1177/8756972821992246.