FROM REGULATION TO AUTOMATION: A FRAMEWORK FOR TRANSLATING REGULATORY REQUIREMENTS INTO EXECUTABLE BUSINESS PROCESSES

Iswarya Srinivasan

Independent Researcher, USA. ORCID ID: 0009-0001-4698-6579

Abstract

Organizations that are subject to high levels of regulation continue to have difficulty expressing legal rules written in natural language as predictable, reliable information systems. The gap between the text of the law and the enterprise software systems used to implement it is a major contributor to failures to comply, high information system maintenance costs, and difficulty keeping pace with change. This article delivers a blueprint for regulatory interpretation as an end-to-end six-stage lifecycle process comprising understanding, formalizing, mapping, validating, monitoring, and updating, as well as a method for the systematic breakdown of regulatory requirements into structured business rules, the mapping of such business rules onto executable business process schemas, and the tracing of their enactment in decentralized multisystem enterprise integrations. By recording cross-system data lineage (correlation identifiers and context propagation), and applying human judgment in those gray areas of the law, the framework ensures that automated processes are visible, controllable and continuously aligned with the intent of regulations. This discussion situates the framework in the context of the current regulatory technology and model-based compliance checking discussions. Finally, it discusses implications for organizations dealing with compliance in interconnected enterprise systems.

Keywords: regulatory technology; business process management; compliance automation; enterprise systems integration; conformance checking

1. Introduction

Regulatory requirements are typically expressed in natural language. Information systems use rules, validation, workflows, data models and controls. Bridging this gap is an active operational challenge with meaningful consequences. Legislative and regulatory requirements can change more frequently than software pipelines can be updated to accommodate the changes. A single change to a regulatory requirement could have downstream implications across many workflows, data elements, reports, or approvals in a single organization. Federal inspectors general, the watchdogs of government agencies, summed up the scale of the problem when they reported that as of fiscal year 2022 only 8 of 23 civilian federal departments and agencies had effective information security programs, a long way away from compliance with regulations [2]. Technology-enabled approaches to improving the transparency, monitorability and adaptiveness of regulatory obligations are increasingly recognized in regulatory technology and business-process compliance scholarship [1, 5, 6].

Generalized practitioner experience from large-scale and regulated enterprise transformations provides an analogy. A business or regulatory requirement must be understood in context, operationalized, reflected in system behavior, validated in representative scenarios, and monitored in production. This framework defines a regulation's interpretation lifecycle: understand, formalize, map, validate, monitor, and update the regulation interpretation.

The present invention solves the above-mentioned problem by abandoning the concept of a single standalone workflow engine, which is appropriate only for legacy single-system environments. Instead, a three-dimensional model is proposed, where the structural pillars of multi-system integration, Business Process Management (BPM) automation and traceable regulatory compliance are equal pillars. This article attempts to describe how a compliance lifecycle can be propagated in distributed and connected environments, in order to support secure and compliant digital automation in the enterprise and public sector.

2. Theoretical Framework and Literature Review

Research is active on the interplay between corporate governance, business-process compliance and information systems design, as manual compliance audits are increasingly unsuitable in view of the speed and intricacy of modern statutory frameworks [1, 6, 18].

2.1 The Evolution of Regulatory Technology

Regulatory technology (RegTech) emerged from the literature on manual, checklist-driven compliance management. Historically compliance was a defensive, retrospective activity, focused on creating documentation, and was reviewed periodically at set time intervals. Documented processes did not have a real-time nature. Recent work argues that this move to a proactive compliance architecture can lead to reduced operational and reputational risk and increased reliability [1, 18]. Accomplishing this requires moving from rule sheets described in natural language to machine-readable policy representation, which in the literature is sometimes referred to as policy-as-code, where the compliance reasoning is represented in a way that is directly readable by software and not separately interpretable by a human [16].

This is important because in most enterprise environments, the volume and velocity of regulatory change exceeds the capacity of manual review to keep up. A quarterly compliance checklist is not sufficient for an environment in which individual regulators are regularly issuing amendments, guidance updates, and enforcement clarifications throughout the year. Applied work indicates that policy-as-code tooling can create and validate machine-readable compliance policies from infrastructure and configuration definitions, thereby reducing the gap between creating the policy and enforcing it in production systems [16]. In other words, the difference is not simply that it becomes more convenient to apply, but rather that the risk is shifted from being audited after the fact, to being an enforced property of the system on a continuous basis.

2.2 Computer Code Versus Statutory Intention

Another long-standing issue in law and language is that of replacing law with computer code [3]. If lawyers write textual rules for coders, the intent of the legislation is not likely to be faithfully represented in the code unless there is a shared language. One reason for this is that business process modeling has traditionally not treated compliance checking as a first-class design requirement [4]. However, recent years have seen the emergence of general model-based conformance checking, where the behavior of the automated system is checked at runtime against a separate, explicit and verifiable representation of the underlying legal requirement [4, 5, 7].

2.3 Model-Based Conformance and Compliance Checking

Model-based business-process conformance checking is relatively well-explored and guided by a well-developed body of knowledge, following the early work, which identified fragmentation and a lack of shared benchmark problems in the field [4]. More recent work has produced taxonomies of regulatory compliance requirements for business processes that connect regulation characteristics to conformance-checking techniques that can enforce them [5]. Surveys of monitoring for regulatory compliance have also examined the entire space of regulatory compliance monitoring techniques, highlighting that this work has focused on tool-supported but not automated monitoring of regulatory compliance, rather than automated and real-time monitoring [7]. A similar shift towards automation and large language models has been seen in research on natural language processing for requirements engineering. These tools focus on the automatic detection of ambiguity and inconsistency in natural language requirements [8]. Taken together, this indicates that regulation-to-system translation is best thought of as a process rather than simply an interpretation task performed at the beginning of system implementation.

Two other streams of this literature are also of potential interest to enterprise practice. The empirical analyzes of process-mining and compliance-checking adoption in large enterprises or compliance issues within regulated domains indirectly suggest that the gap between what techniques can show in research and what commercial tooling can deliver in practice remains meaningful. However, many of the identified compliance support features need to be manually configured or interpreted before they can provide support for a specific compliance requirement [6]. Second, research on BPM as an enabler of digital transformation suggests that organizations will only realize the benefits of automation when they are taking a combined approach to process redesign and compliance rather than a compliance-led approach towards defining processes [6]. Again, these two examples suggest that in a framework that is oriented towards the realization of the process, the checking of compliance logics should rather be integrated in the process design activity as such than being a verification step.

3. The Regulation-to-System Translation Gap

While regulations are typically written in human-readable form with requirements, exceptions, thresholds, rights and responsibilities, and reporting obligations, software requires more precise codifications. Ambiguity exists in cases where a requirement is open to interpretation, where exceptions to rules are context-dependent, or where reasonable interpretations vary among business

units. These requirements are magnified further when a single regulation impacts multiple interconnected systems managed by different teams.

The size of the gap is indicated by the scale of the federal government's IT activities. US civilian departments spend over $100 billion on IT each year, a large part on defenses to operate and maintain the current IT infrastructure, not just in order to build new IT [20]. When regulatory change must be threaded through systems of this scale and age, informal or undocumented translation of legal text into system logic is a major source of operational and compliance risk.

The first control against this risk is structural requirement decomposition. Each requirement can be decomposed by actor, trigger, condition, action, exception, evidence and reporting consequence, see table 1. This provides a clear reference from the originating requirement to the business process that results from it. A practitioner who works as a liaison between business and technology teams is well positioned to recognize areas in the policy language that contain ambiguities that should be resolved before compilation into system behavior. It is easier to address this uncertainty before deployment than after.

Table 1: Regulatory Requirement Decomposition Template (illustrative)

Field ID	Parameter	System Rule	Example
P-01	Actor	Who triggers the rule	Financial officer
P-02	Trigger	What starts the process	Transfer above a defined monetary threshold
P-03	Condition	What must be true	Recipient located in a restricted jurisdiction
P-04	Action	What the system does	Pause payment and route for review
P-05	Exception	When to skip the rule	Approved government-to-government transaction
P-06	Evidence	What is retained for audit	Timestamped log record
P-07	Reporting	Who is notified	Automated alert to the responsible oversight function

Note: field values are illustrative and intended to demonstrate the decomposition structure; they do not represent a specific implemented system.

4. The Cross-System End-to-End Lifecycle

To solve this translation gap, compliance is reflected in a continuous cycle under the framework as opposed to being part of the delivery of a single project. The continuous cycle translates to distributed systems such that changing a piece of regulation will propagate to every connected distributed environment rather than just the environment the change was initiated in. The six stages of this lifecycle are also shown in Figure 1.

Figure 1: End-to-end regulation-to-automation lifecycle. Six-phase lifecycle for translating regulatory requirements into executable, continuously monitored business processes.



4.1 Understand

The initial phase is when continuing regulatory monitoring from compliance and legal functions scan for new regulations, statutory changes, and administrative initiatives that are organized through an intake process. Once the change is identified, an impact analysis is performed to identify what applications may be impacted by the legal change through the organization's software asset inventory.

4.2 Formalize

Once a regulatory change is identified, the relevant legal text is restructured according to the rule definition format in Table 1. Policy and compliance experts then analyze the legal text as logical variables, deciding the applicable actors, triggers, and outputs that allow the unstructured legal text to be formed into machine-interpretable structured rule definitions. Natural language processing technology for requirements engineering is improving, including tooling for finding ambiguous or contradictory language in source material. However, even rules the tools produce still need to be checked by someone with subject matter expertise before they can be reliably used in production settings [8].

4.3 Map

Then, the structured parameters are transformed into technical design specifications. Hence, the engineers can use BPMN business process diagrams [12] to model the compliance boundaries. This includes defining system data models and state boundaries, developing integration points in the appropriate enterprise service layers so that the rules can execute in runtime operational environments.

4.4 Validate

Before a rule can be run in a live production environment, it must be quality assured through boundary testing to make sure that the happy path, edge cases, missing data and conflicting statutory exceptions are covered. Sign off is required at this stage to ensure that the technical rule accurately implements the law as originally intended.

4.5 Monitor

When the system is deployed, it actively starts runtime verification, logging events to an append-only audit record of transactional events based on execution telemetry. In this way, live application behavior can be constantly checked against the compliance model, so unexpected runtime exceptions can be detected quickly, rather than only upon retrospective adherence checks.

4.6 Update

During the update phase, the lifecycle stops and then restarts. This phase handles features that are constantly updating. If a regulator adds or changes a threshold, or some other operating condition, the resulting delta update adds a flag to each rule module that could be affected, runs isolated regression tests, updates the audit traceability record and assumes that the system will adjust without disrupting any other business operations.

5. Multi-System Traceability and Propagation Architecture

One of the more common failure modes with enterprise software is the assumption that one workflow engine can handle the whole compliance lifecycle when, as a rule, workflows typically span multiple integrated systems. For example, a benefit payment may require a front-end portal, an identity verification provider, an eligibility engine, a financial ledger, and an external banking interface, all of which may be independently developed and have different release cycles.

To address that, the framework proposes three complementary mechanisms that will allow traceability to be propagated.

5.1 Global Correlation Identifiers

All transactions include a correlation ID, which is created when the transaction is initially ingested and then passed in the metadata payload as it transitions through systems, such as an identity verification service to a financial engine. This allows an auditor or investigator to retrace the path of a single transaction across the various networks, applications and databases that it has passed through, without having to retroactively correlate log entries between different system components.

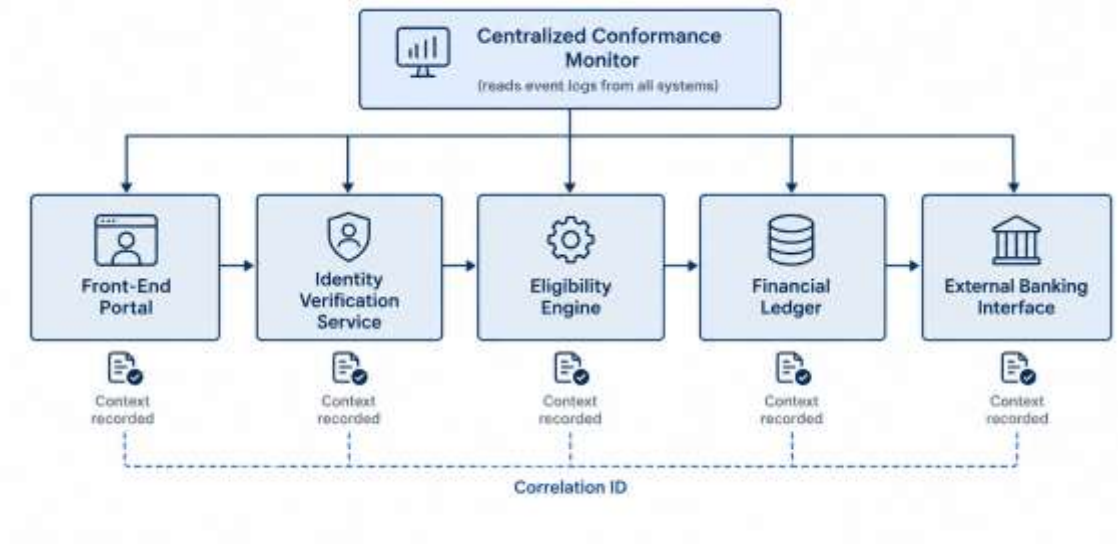
5.2 Distributed Context Propagation

When an integrated system calls a business rule, it sends the version of the business rule, along with the inputs and outputs, to the next system. This preserves regulatory compliance context information across system boundaries and avoids a class of error where several systems apply stale or conflicting versions of the same business rule on the same transaction during a transition time where the regulation has been updated.

5.3 Integrated Conformance Checking

This is done by a central monitoring function that continuously reads the event logs of these distributed systems and compares them against the central compliance model to identify process violations or other deviations occurring anywhere in the network. This is similar to the state of the art in conformance checking, which has more recently shifted towards real-time automated monitoring as a complement to, rather than a replacement of, periodical manual checking, although most of this work so far has relied on tools [5, 7].

Figure 2: Multi-system traceability and context propagation architecture. Correlation identifiers and centralized conformance monitoring preserve compliance context as a transaction moves across independently maintained systems.



6. Business Process Automation and Human-in-the-Loop Governance

Business process management makes it easier to capture compliance controls in a human-readable way. This is due to the presence of activities, decisions, roles and transitions, as well as the explicit locations where controls should be applied, as opposed to implicit encoding of controls in code [4, 13]. This reduces reliance upon informal interpretation and can improve consistency of understanding across users, teams, and locations.

Automation's limitations can be justified when the regulation contains genuine ambiguity or requires contextual judgment. A more refined model distinguishes deterministic rules that can be safely automated from exception cases that require structured human review and policy cases that should be sent to a subject-matter authority. This human-in-the-loop design is consistent with recent work that argues human involvement should be preserved at decision points where automated rules could otherwise produce inappropriate or biased outcomes, in order to ensure higher compliance rates, preserve accountability, and prevent rigid automation from producing outcomes that would otherwise be viewed as inappropriate [17].

The design of the escalation path should be given careful attention. A case escalated to humans should, at the very least, include the rule threshold at which the case was escalated, the input data being analyzed, and the output data from the previous case, so that the human does not need to 'go detective' on the raw system log. Otherwise the review queue is a bottleneck rather than a safety valve because every case is created much faster than it can be meaningfully reviewed. A properly designed human-in-the-loop layer in the compliance life cycle does much better. The reviewer is engaged in active learning, and the feedback from every escalation is used to modify the rules defined in the formalize phase in Section 4 until the uncertainty is resolved with a modified rule rather than a collection of human reviews.

7. Verification and Audit Matrix

Traceability matrices can ensure that all laws embedded across systems can be traced back to the laws themselves, the rules, test cases, and audit records that determine compliance and support the enforcement of these rules (Table 2). However, systematic reviews of regulatory compliance monitoring have determined that traceability matrices are not often created compared to the number of techniques reported, indicating a gap between research-based techniques and production use [7]. Using a matrix of this kind is one practical method of addressing that gap in a single organization, even if full automation of every control being mapped is not yet possible.

Table 2: Multi-System Regulatory Traceability Matrix (illustrative example)

Source Requirement	Target System Function	Executable Rule Type	Audit Evidence Type
Access control requirement (e.g., restricting unauthorized access to sensitive records)	Identity and access gateway	Authentication and authorization scope rules	Access log record
Encryption requirement (e.g., protecting data at rest)	Storage layer	Cipher configuration policy	Key management configuration record
Electronic record integrity requirement (e.g., unalterable signatures)	Case management system	Append-only signature ledger	Ledger hash receipt
Cross-border transaction reporting requirement	Payment processing ledger	Threshold-based rule engine check	Transaction audit stream entry

Note: requirement categories and system functions are illustrative and generalized; they are not drawn from a specific implemented system or named regulation-to-control mapping.

8. Strategic and Societal Implications

The impact of a framework such as this goes beyond a reduced effort to comply with rules by hand. Under multi-system environments, automating compliance logic across these boundaries can greatly reduce the cost of regulatory compliance for public authorities and private sector parties while delivering services to citizens and businesses faster.

Clear end-to-end data lineage across integrated enterprise systems can lead to more efficient and effective audits. If the correlation identifier and its context can be followed through the life of each transaction across every system it touches, there is less need for auditors to view and reconstruct the story of the transaction from multiple logs.

Finally, where automated compliance validation is embedded into cross-system data pipelines, it gives some measure of durability to the other parts of the digital infrastructure: systems that perform compliance validation against an explicit model of permissible behavior are at least in principle more likely to discover errant or unauthorized behavior than systems which have to rely on manual verification of the model at intervals, so long as the model is reasonably well maintained.

And there are implications for organizations operating legacy or aging infrastructure, where the case for structured compliance automation is often greatest: oversight reporting on federal information technology has identified a set of critical legacy systems that multiple agencies are still using, despite their known cybersecurity exposure (showing how difficult outright replacement can be, even where a case for modernization can be made) [20]. Organizations likely to face a similar combination of old systems and continuous regulation are unlikely to be able to replace the entire set within a reasonable time frame, or before the next regulatory change in almost all cases. It is more realistic to have a framework be layered on top of existing systems rather than have it replace existing systems altogether.

9. Conclusion

This article outlines a three-dimension framework to map regulatory text to auditable and automated business processes. It seeks to maintain auditable business processes that comply with legal requirements over time, balancing multi-system integration, BPM automation and continuous documentation of compliance in a traceable manner over time.

This distributed traceability model with global correlation identifiers and active context propagation (described above) offers a way of enabling compliance tracking across complex, multi-system networks in a way that is no longer dependent on a single, centralized workflow engine. Meanwhile,

the remaining human actions in these networks continue to support professional accountability, even when regulatory frameworks favor fully automated processes.
Performance data released by federal regulators on the effectiveness of the programs and IT modernization backlog show that the translation problem that this framework addresses is not only theoretical, nor limited to one domain [2, 20]. Future research might study whether natural language processing and process mining approaches can further automate the formalize and monitor stages of the lifecycle, while preserving the human discretion that regulatory uncertainty often requires.

Ethics Statement
Conflict of Interest
The author declares no conflict of interest.

Statement of Human and Animal Rights
This article does not involve any studies with human participants or animals performed by the author.

Informed Consent
This article does not contain any studies requiring informed consent from human participants.

Declaration of AI Tool Use
No generative AI tools were used to generate the scholarly content of this manuscript. The author affirms responsibility for the entirety of the content.

References
[1] Grassi, L., Lanfranchi, D. RegTech in public and private sectors: the nexus between data, technology and regulation. Journal of Industrial and Business Economics, 49, 441-479, 2022.
[2] U.S. Government Accountability Office. Cybersecurity: OMB Should Improve Information Security Performance Metrics. GAO-24-106291, Washington, D.C., 2024.
[3] Micheler, E., Whaley, A. Regulatory technology: replacing law with computer code. European Business Organization Law Review, 21(2), 349-377, 2020.
[4] Becker, J., Delfmann, P., Eggert, M., Schwittay, S. Generalizability and applicability of model-based business process compliance-checking approaches. BuR Business Research, 5(2), 221-247, 2012.
[5] Grohs, M., Klessascheck, F., Janicke-Helling, A., Pufahl, L., Rehse, J.-R. Can conformance checking monitor process compliance? A taxonomy of regulations for business processes. Process Science, 3(1), article 11, 2026.
[6] Adams, N., Augusto, A., Davern, M., La Rosa, M. Addressing the contemporary challenges of business process compliance. Business and Information Systems Engineering, 68, 377-400, 2026.
[7] Klessascheck, F., Pufahl, L. Reviewing uses of regulatory compliance monitoring. Software and Systems Modeling, 25, 413-436, 2026.
[8] Necula, S.-C., Dumitriu, F., Greavu-Serban, V. A systematic literature review on using natural language processing in software requirements engineering. Electronics, 13(11), 2055, 2024.
[9] Barrientos, M., Winter, K., Rinderle-Ma, S. Impact analysis of regulatory requirement changes on business process compliance. Information and Software Technology, 194, 108079, 2026.
[10] Mustroph, H., Barrientos, M., Winter, K., Rinderle-Ma, S. Verifying resource compliance requirements from natural language text over event logs. In: Business Process Management (BPM 2023), Lecture Notes in Computer Science, vol. 14159, 249-265, Springer, Cham, 2023.
[11] Winter, K., van der Aa, H., Rinderle-Ma, S., Weidlich, M. Assessing the compliance of business process models with regulatory documents. In: Conceptual Modeling (ER 2020), Lecture Notes in Computer Science, vol. 12400, 189-203, Springer, Cham, 2020.
[12] Object Management Group. Business Process Model and Notation (BPMN), Version 2.0.2. OMG Document formal/2014-01-01, 2014.
[13] Putra, H., Mahendrawathi, E.R. The role of business process management in digital innovation and digital transformation: a systematic literature review. Procedia Computer Science, 234, 829-836, 2024.
[14] Li, B., Peng, X., Xiang, Q., Wang, H., Xie, T., Sun, J., Liu, X. Enjoy your observability: an industrial survey of microservice tracing and analysis. Empirical Software Engineering, 27, article 25, 2022.
[15] Liu, Z., Wu, H. A framework for dynamic blockchain-based data auditing. International Journal of Accounting Information Systems, 56, 100737, 2025.

[16] Romeo, F., Arena, L., Blefari, F., Pironti, F.A., Lupinacci, M., Furfaro, A. ARPaCCino: an agentic-RAG for policy as code compliance. In: New Trends in Database and Information Systems (ADBIS 2025), 467-481, Springer, Cham, 2026.
[17] Kappel, M., Neuberger, J., Mohrlein, F., Weinzierl, S., Matzner, M., Jablonski, S. A human-in-the-loop approach for improving fairness in predictive business process monitoring. In: Business Process Management (BPM 2025), Lecture Notes in Computer Science, vol. 16044, 343-360, Springer, Cham, 2026.
[18] Dziawgo, T. Regulatory Technology (RegTech) as an enhancement of compliance processes in financial institutions. Copernican Journal of Finance and Accounting, 14(2), 27-43, 2025.
[19] National Institute of Standards and Technology. Security and Privacy Controls for Information Systems and Organizations. NIST Special Publication 800-53, Revision 5, Gaithersburg, MD, 2020 (updated 2024).
[20] U.S. Government Accountability Office. Information Technology: Agencies Need to Plan for Modernizing Critical Decades-Old Legacy Systems. GAO-25-107795, Washington, D.C., 2025.