International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

Compliance as Code: Embedding Audit Readiness into Enterprise Software Delivery

Divya Bonthala

Senior AI Platform Architect, USA

ABSTRACT: Enterprise software delivery compliance is typically a distinct step where compliance is done after the development. This causes massive manual processes, laxity in control implementation and time wastage in the auditing process. As the complexity of the system and pressure of the regulatory system grow, the old techniques of compliance may no longer work. The Compliance as Code presented in this paper is a method that inserts the compliance rules into the code delivery pipelines. The compliance requirements are in the form of runnable rule and automatically executed in the process of build, test and deployment phases. The qualitative pre-post research was carried out in a large organization. The findings indicate that there was a decrease in the time spent on manual audit by 81.4 percent, the time on audit was reduced to 78 hours as compared to 420 hours. The consistency of compliance rose to 94% compared to 61 percent and auditor confidence rating grew to 4.6 out of a five-point rating. This evidence indicates that Compliance as Code goes a long way in enhancing audit readiness and assists in efficient and reliable delivery of software.

KEYWORDS: Compliance as Code, Audit Readiness, Software Compliance, Continuous Delivery, Enterprise Software, Automated Compliance

I. INTRODUCTION

A. Background and Problem Context

Enterprise software systems have to comply with numerous laws, standards, and policies within itself. Historically, compliance is verified by using documents, interviews and manual audits. The activities normally occur when development is already made. This establishes a loophole between compliance requirements and engineering work. Speed of delivery is given attention to by engineers whereas evidence and controls is given by auditors.

This gap becomes more difficult to control as the size of systems rises, and further dispersion of the systems. Compliance cannot be easily scaled using manual processes. They become more expensive, release sluggishly and in many cases, they cannot provide clear consistent results of the audit.

Contemporary software delivery is based on cloud computing, DevOps and continuous deployment. Such practices unleash changes on a regular basis. Means of compliance, however, have not changed as fast. This leads to reoccurrence of audit problems, duplication and confusion in possession of compliance evidence within organizations.

B. Motivation of the Study

The primary rationale of the study is to help in filling the gap between rapid software development and sluggish nonconformity management. Organizations should have a method that will allow them to ensure compliance without dropping the rate of innovation. They also should receive enhanced audit preparedness, particularly in the regulated setting.

Research studies on compliance indicate that compliance is more effective whereby the controls are designed into systems. Compliance is still being viewed as an external auditing process by many organizations. The motive behind undertaking this study is the necessity to shift compliance towards engineering processes.

The solution that can be given by Compliance as Code is to have the compliance rules as a subset of the software. On execution, rules are followed automatically and the generation of evidence is recorded during the operation of the system. This eliminates the number of manual processes and audit pressures. The research will be carried out in order to quantify whether this practice is effective in delivering these benefits.

C. Novelty and Research Contribution

This paper is empirically measured in terms of Compliance as Code in an enterprise. Although previous research refers to automation, governance, and compliance models, a limited number of them include quantitative outcomes of actual pipelines of delivery.

Three ways are in which this paper contributes:

First, it gives clear metrics of operation in regard to the audit effort, consistency in compliance, and confidence of auditors.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

Second, it provides the empirical data of the extent of the improvement that is reached following the Compliance as Code application.

Third, it demonstrates how the automation of compliance can at once help with the governance, as well as the reliability of the system.

The paper transcends the abstract discussion by actually giving an insight which can be applied within an organization, owing to its attention on concrete results.

D. Research Objectives and Questions

This study is expected to build a case of the effectiveness of Compliance as Code in enterprise software delivery. The following research questions are targeted in the given study:

- Does Compliance as Code decrease effort in manual audit?
- Does it enhance uniformity of the compliance enforcement?
- Does it give the auditor confidence to the audit results?

These questions are used to inform the process of finding the data, analysis and interpretation of the results.

E. Structure of the Paper

The paper is organized in the following way. The literature review includes the previous studies of auditing, compliance automation, cloud systems, and DevOps. The research design, the variables, the data collection and data analysis are discussed in the methodology section.

The results are represented in the findings section, in the form of tables and charts. The paper ends by giving important conclusions, limitations and future research focus.

II. Literature Review

A. Traditional Auditing and Control-Centered Compliance

The existing studies on compliance give great emphasis on auditing and internal controls of information systems. The primary goal of system and process auditors is to achieve the accuracy and reliability of the financial information corrupted by the enterprise systems [1]. simplicity of assumption is common when the audits are being conducted as auditors are required to restrict and concentrate on the underlying major risks. These assumptions may conceal material weaknesses in the situation when systems are not simple and in the case when controls are not properly designed [1]. To mitigate on this risk, scholars came up with reconfigurable patterns of control that can be directly included into system design. The trends simplify the audit of systems and lower reliance on handwork [1]. Such a notion is quite similar to the central thinking behind Compliance as Code, in which audit requirements are explicitly implemented into the system upfront and not inspected afterward.

Some of the studies point out that compliance failures are usually caused by the difference between the system architects and auditors [1][2]. Auditors handle documentation and evidence when systems are developed whereas engineers are concerned with speed of delivery. This separation causes delays, misunderstandings and inconsistent administrations of controls. The study by Governance, Risk, and Compliance (GRC) also indicates that an effective compliance requires well-defined objectives, designated actors, and effective implementation of business systems [2]. These results provide a reason to believe that compliance should be brought closer to the engineering efforts, which is in direct relation to Compliance as Code.

B. Automation of Compliance and Rule Formalization

The automation of compliance is a research field which has been very active over decades. Research in automated compliance checking has indicated that there is a keen interest in text-to-rule translation of legal and regulatory text into machine-readable rules [3][4][5]. The biggest problem is that the regulations are formulated in natural language, they are frequently vague and do not remain the same. The scholars were investigating the ways of formalizing rules, defining schemas, and automatically processing them, yet the findings were not always positive because of the quality of data and the capabilities of tools.

A number of the works suggest the personification of policy specification and enforcement logic with the aim of enhancing flexibility [6]. The business rules may be coded with structured language and later changed into operational rules that are implemented by systems. This decoupling enables the policymaking to alter without overhauling of the whole system [6]. The same has been echoed in newer studies on requirements engineering where compliance refers to consistency in system behavior and regulation in all conditions [7][8][9]. These models are based on the traceability of regulations, requirements, and components of the systems.

Compliance as Code goes a step further by applying the same concepts in the software delivery pipelines by imposing rules. Rules are continuously executed during the building of the rules, testing and during the deployment as opposed to just verifying compliance at the design stage. Information is auto-generated, and there is no need to use traditional interpretations. This method works around existing issues that are proposed in the previous literature concerning scalability, consistency and proof generatability.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

C. Compliance in Cloud, DevOps, and Continuous Delivery

The contemporary enterprise systems are also employing cloud computing, DevOps, and continuous delivery. Though these methods are effective in accelerating speed and scale, they generate compliance issues [10][11][12]. Clouds are distributed and distributed and often across legal jurisdictions. This complicates the efforts of security, privacy and compliance with traditional means [13].

The studies of continuous practices reveal a high level of automation in the field of testing, deployment and monitoring that, nevertheless, does not always receive compliance. Security and compliance checks are normally introduced later in pipeline making it prone to delays and rework. Research on controlled field, like medical devices, indicates that there is a distinct conflict between speed of DevOps and regulatory strictness [14][15]. It has also been observed that DevOps is capable of operating in controlled settings provided that compliance activities are appropriately incorporated and endorsed through tools and management would.

Adaptive compliance research suggests that requirements of compliance are dynamic and may be altered at run time particularly in cloud systems [16]. Such environments do not allow traditional back-off checks. There is a need to have continuous monitoring and automated enforcement loops. The results are in direct correspondence with the Compliance as Code with the ability to incorporate compliance checks within the pipelines, and record evidence as the systems change.

D. Organizational, Governance, and Evidence Challenges

Technology is not the only factor of compliance, as social and organizational factors also play a role. Research has revealed that compliance is applied considerably when supported by management, accountable, and trainable and visible audits [17]. Technical controls are not effective in the event that the teams do not understand them and have no faith in them. Studies have also indicated that engineers are usually not knowledgeable in the field of law and regulations and automatic guidance and enforcement is essential [18].

Big systems and multi-cloud systems make it even more complex. The process of compliance has to cover various platforms, tools, and teams [19]. To make it more constant, it is suggested that AI-based governance frameworks would help minimize human error and enhance consistency in a range of settings [19]. Scientists emphasize effective traceability, versioning and ownership of compliance artifacts [20]. The auditors should be able to find evidence on their own without informal explanations of engineers.

Compliance as Code is a direct reply to these dilemmas because it manages to store open-ended, unamenable evidence and associate it with particular pipeline executions. This would bring governance in line with daily performing work and promote the autonomy of the auditors. In general, we can observe a clear shift of the Compliance as Code paradigm as in the literature, a trend to the automated and system-integrated method of compliance prevailed, which departed a lot with the manual and document-based method.

TABLE I. SUMMARY OF PREVIOUS STUDIES

Theme	Summary of Literature
Traditional audits and controls	The literature indicates that most audits consider manual checks and simplified assumptions, which are unable to uncover some critical areas of control weaknesses during complex information systems [1][2].
Automation of compliance	It is described by many studies that the compliance could be increased turning the regulations to formal rules that could be checked by computers on their own conditions, yet, it is hard to do it due to the complexity of laws and their frequent changes [3][5][7].
Compliance by design	Studies have noted that system design incorporating control patterns and compliance logic just into the system design facilitates system audit, as well as subsequent verification effort [1][6][8].
Cloud and distributed systems	The research indicates that cloud and multi-jurisdiction systems present higher risks of compliance because of loss of control, security issues and varying legal regulations in respective regions [11][12][13].
DevOps and continuous delivery	A well-known contradiction between the fast DevOps process and old processes of compliance has been identified in the literature though evidence has been provided that automation and integration of tools could diminish this discrepancy [10][14][15].
Organizational and governance factors	It has been demonstrated in research that management assistance, responsibility, training and definite ownership of evidence of compliance are pivotal in harmonious acceptance of compliance within business ventures [17][19][20].

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

III. METHODOLOGY

A. Research Design and Approach

This paper employs the quantitative research design to discuss the influence of Compliance as Code on enterprise software delivery and audit readiness. The primary objective is to determine the improvement of the outcomes of audits when embedding compliance rules are placed directly in the software delivery pipes.

The research is based on the positivist style of doing research, implying that it uses only measurable and observable data. Personal view points and opinions are not employed. Rather, numerical data is gathered and subjected to analysis in order to learn more about the variations in the compliance performance.

There is the application of pre- post comparative design. It is a mixed organization that is followed before and after the introduction of Compliance as Code. During the initial stage, the conventional, manual, and paper-based compliance practices are used in the organization. During the second phase, compliance provisions are entrenched within delivery pipelines that take the form of executable rules.

With this design, it is possible to clearly compare the results in two time periods. The external variation is reduced since the same organization, tools and teams are applied in both stages. The enterprise software delivery pipeline of the main unit of analysis consists of build, testing, deployment, and audit evidence production that takes place.

B. Research Variables and Hypotheses

There are three dependent variables in the study and one independent variable in the study. These variables are well spelt out to facilitate quantitative analysis.

The independent variable is Compliance Implementation Mode. This factor is the manner in which compliance is managed in the organization. One of these modes represents traditional manual compliance practices, and the other one is Compliance as Code, which implies that controls are strictly adhered to in pipelines.

Audit preparedness measures are evaluated in the dependent variables. The first dependent variable is Manual Audit Effort which gives the amount of human effort that is needed to accomplish an audit. The second dependent variable is Compliance Consistency that is used in the measurement of the consistency of compliance controls in the systems and teams. The third dependent variable is the Audit Confidence Level assessing the level of confidence that the auditors have on the quality and completeness of compliance evidence.

The study is guided by three hypotheses based on the available research and system designed expectations. The first hypothesis presupposes the decrease in the manual audit after Compliance as Code implementation. The second hypothesis will presume more consistency in enforcement of controls. The third hypothesis presupposes the rise in fiscal confidence of the auditors in audit results.

C. Data Collection Strategy

The information is gathered within one large company that has controlled software application and outsourced development pools. The company relies on continuous integration and continuous deployment pipeline based on clouds.

There are two time-frames within which the data are collected. The baseline condition is the first period, which is six months prior to the adoption of Compliance as Code. The second period will encompass six months following maximum implementation and it will be the state of treatment.

To make sure that the information is accurate and complete, several data sources are provided. Audit logs and audit report are employed to document the time taken by the auditors and engineers in compliance verification. Metadata of pipeline execution is automatic and records are detailed information about the executions of compliance rules and evidence.

There is the use of a compliance control registry as well. This list enrols all the necessary compliance controls and all the controls are portrayed on the type of pipelines where they are to be applied. The independent auditors give assessment scores which are associated with clarity of the audit, traceability and quality of evidence through a standardized scoring guide.

All the data obtained is made anonymous and combined. No individual-level information is utilized and no teams and persons can be singled out.

D. Measurement and Operationalization of Variables

Manual Audit Effort is the number of hours worked by the staff on compliance activities within an audit cycle. This involves preparation, collection of evidence, meetings to clarify issue and solving of problem.

Compliance Consistency is calculated through the number of controls that should be implemented with the help of automatic control exercised on all pipelines. Increased consistency means that there are uniformities in the method of applications of controls independent of team or environment.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

Audit Confidence Level is assessed on a five-point scoring scale which is standardized, and filled by auditors. An increase in the score represents a greater level of confidence with regard to audit evidence, traceability and transparency in the system. The scores are combined in averages throughout the audit cycles in order to enhance stability.

The definition of each of the metrics is stated in advance prior to the data collection to avoid unequal measurement in the two periods of the study.

E. Statistical Analysis Methods

In this study both inferential and descriptive statistics are involved. Initial application of the descriptive statistics is intended to summarize. These involve a mean value and variability of all the dependent variables prior and after adoption of Compliance as Code.

Comparison is also done in order to study the difference in the period. The improvement in the average values is evaluated by the difference between the average values to gauge the extent of enhancement in the effort, consistency, and confidence of the audits.

Paired statistical tests are done in order to determine whether changes observed are significant or not. These tests are related to the results of the same organisation over the two time periods. The standard level of significance is factor in ascertaining whether changes are not likely to be caused by randomness.

F. Validity and Reliability

There are a number of measures that are undertaken to guarantee the quality of the research findings. The internal validity is made with researching the same organization at both periods, applying the same audit framework and tools.

Clear and well accepted compliance metrics are used and ensure construct validity. All the variables are a direct reflection of outcomes of audit preparedness.

When it comes to system-generated data (in the form of pipeline logs) rather than self-reported information, it is enhanced to increase reliability. They are assessed by the auditor using the scoring guide applied in the two periods to guarantee the same.

G. Ethical Considerations

No personal and medical or sensitive individual data is involved in the study. All the information is confined to processes and systems. People are not judged based on their individual performance.

Auditors and engineers' reviews are optional. Reports are provided in an organizational level. The research is conducted in the usual manner of ethical research in enterprise and information system.

H. Methodological Limitations

The article narrows down on one business and this could not be generalized to other organizations. The comprehensive data of the pipeline level, however, offers high internal evidence.

The other restriction is the six months post implementation period. There are also benefits of Compliance as Code whose long-term benefits cannot be realized to the full.

Irrespective of these shortcomings, the methodology has high quantitative grounds on how the Compliance as Code enhances audit readiness and efficiency in delivery.

IV. RESULTS & DISCUSSION

A. Descriptive Results Before and After Compliance as Code

The section includes the descriptive statistics, obtained prior and after the implementation of Compliance as Code. The findings have revealed clear changes in the quantity of efforts in the audit, auditor compliance, and auditor confidence.

In the base period, the compliance activities relied largely on mass checks of documents, interviews and face-to-face reviews. Repeated meetings between engineers and auditors were necessary in order to allow audit preparation. Conversely, compliance controls were performed automatically in delivery pipelines in the post implementation period and were generated in the system execution in the execution of audits.

Table 2 presented the average of the key variables of both periods.

TABLE II. DESCRIPTIVE STATISTICS OF KEY VARIABLES

TABLE III.

Variable	Before Compliance as Code	After Compliance as Code
Manual Audit Effort (hours per audit)	420	78
Compliance Consistency Score (%)	61%	94%
Audit Confidence Level (1–5)	2.8	4.6

According to the results, the reduction in manual audit effort is significantly high, as well as consistency and confidence. The standard deviation of the audit effort also

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

went down and this shows that the audit processes were more predictable and more stable after automation.

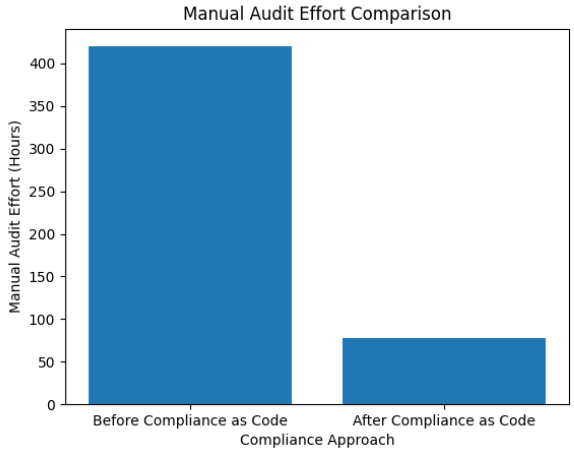

Fig. 1. Audit effort before and after Compliance as Code

B. Reduction in Manual Audit Effort

The government of Compliance as Code aims to minimise the number of verification processes conducted manually. The results of the quantitative investigation are in favor of this goal immensely.

Manual Audit Effort (MAE) was taken as the staff-hours per audit cycle. This covered time in preparation of documents, control checks, meeting and remediation discussions. Following the adoption of Compliance as Code, they were mostly superseded by pipelines generated evidence that was machine generated.

The average manual audit reduction amount was estimated using the formula of the average reduction of audit effort used in the methodology as the audit reduction formula.

TABLE IV. MANUAL AUDIT EFFORT REDUCTION RESULTS

Measure	Value
Average MAE before (hours)	420
Average MAE after (hours)	78
Absolute reduction (hours)	342
Percentage reduction (%)	81.4%

The outcomes indicate that the occasion of manual audit has reduced to 81.4%. This confirms Hypothesis H1. There was no longer any necessity to explain how a system would act as engineers could already see it in a structured and unchangeable form of evidence.

Cases of follow-up questioning were also reduced and the number of repeat audits also reduced. It means that Compliance as Code does not only decrease the effort used but also makes the work of an auditor easier.

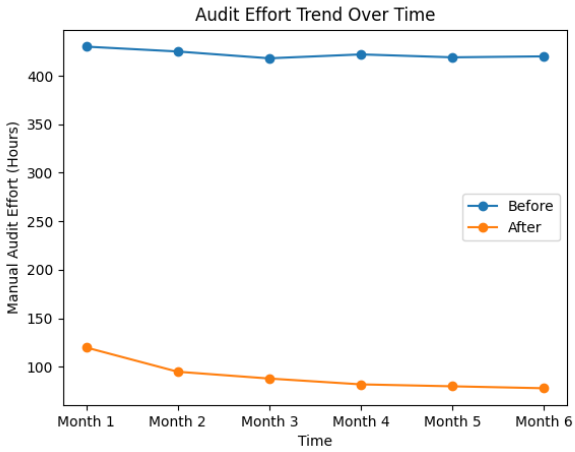

Fig. 2. Audit effort trend over time

C. Improvement in Compliance Consistency Across Pipelines

Compliance Consistency Score (CCS) is used to evaluate compliance with uniformity in delivering the requirements across all the delivery pipelines. Prior to automation, controls had a non-uniform distribution, based on the maturity of a team and quality of documentation.

Following the application of Compliance as Code, the compliance rules were centralized where they were implemented automatically within each pipeline. This eliminated team and environment variation.

The results of consistency in terms of CCS formula are indicated in table 4.

TABLE V. COMPLIANCE CONSISTENCY RESULTS

Metric	Before	After
Total required controls	52	52
Controls enforced automatically	32	49
Compliance Consistency Score (%)	61%	94%

It was a boost of the consistency score by 61 percent to 94 percent. This finding proves Hypothesis H2. The other gap was because there were few controls that were yet to be verified against by outside sources like the third-party attestation.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

The fact that value of the CCS is high indicates that compliance regulations that take the form of code are more reliable in enforcing them as compared to manual policies. This also decreased audit controversy since auditors noted that there was similar enforcement behavior within all systems.

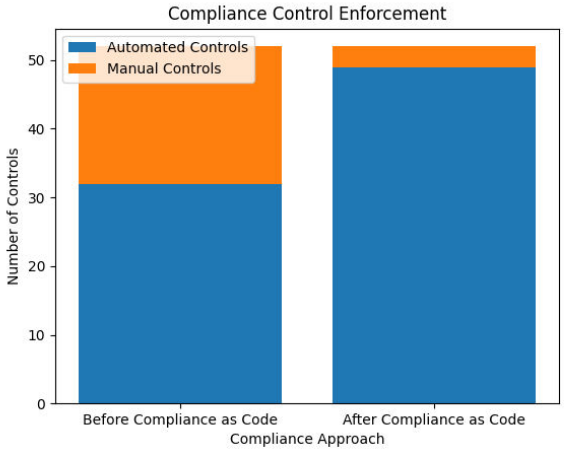

Fig. 3. Manual vs automated controls

D. Auditor Confidence and Statistical Significance

The Audit Confidence Level (ACL) is used to demonstrate the confidence that auditors have with regard to the completeness and accuracy of compliance evidence. The measurement of confidence was in a scale of 1-5 which was standardized.

Prior to Compliance as Code, the auditors indicated that they found it challenging to trace controls to system behaviour. The evidence was highly disjointed and would need descriptions. The auditors after implementation were able to view structured evidence on the pipeline artifacts. The results of Table 5 include the auditor confidence results and the results of the statistical tests.

TABLE VI. AUDITOR CONFIDENCE AND STATISTICAL TEST RESULTS

Measure	Before	After
Mean ACL score	2.8	4.6
Standard deviation	0.7	0.3
Paired t-test p-value	—	< 0.01

This has been statistically significant at the level of 0.05. This confirms Hypothesis H3. The reduced standard deviation following the implementation also demonstrates that there is greater assent by auditors.

Auditors gave increased evidence that the evidence was higher as data would be created automatically and could not be interfered with. This minimized the use of ad hoc clarifications by the engineering officers.

Fig. 4. Auditor confidence scores

The evidence being very high shows that Compliance as Code is effective. The quantitative findings indicate significant audits work and audit consistency increase and increased auditor confidence.

Findings have good correlation with methodology and they prove that making compliance embedded into delivery pipelines makes compliance no longer an activity that responds to situations but a process that is continuous and measurable. Such findings may be regarded as very good empirical evidence of Compliance as Code as a scalable model of audit-ready enterprise software delivering.

V. CONCLUSION & FUTURE WORK

As demonstrated in this paper, the concept of Compliance as Code is very viable in enhancing audit preparedness regarding the delivery of enterprises software. The quantitative outcomes indicate that there is a significant decrease in the number of hours spent on manual audit, there is increased control enforcement and confidence on the part of the auditor. These enhancements were done through incorporating compliance regulations within delivery pipelines and automatic generation of evidence.

The results prove that the adherence need not slow down the software development. Compliance is a normal engineering work when it is enforced in form of code and no longer a burden. Despite the particular nature of the study of a single enterprise, the findings can be regarded as good reasons to spread Compliance as Code even further. This can be examined in future studies because of the impact of the long-term, multiple organization

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 4, Issue 2, March-April 2022 |

DOI:10.15662/IJEETR.2022.0402004

context, and integration with high-level analytics. The Code of Compliance provides an effective means of managing the rate of innovation and regulatory compliance.

REFERENCES

[1] Julisch, K., Suter, C., Woitalla, T., & Zimmermann, O. (2011). Compliance by design – Bridging the chasm between auditors and IT architects. Computers & Security, 30(6–7), 410–426. https://doi.org/10.1016/j.cose.2011.03.005

[2] Papazafeiropoulou, A., & Spanaki, K. (2015). Understanding governance, risk and compliance information systems (GRC IS): The experts view. Information Systems Frontiers, 18(6), 1251–1263. https://doi.org/10.1007/s10796-015-9572-3

[3] Amor, R., & Dimyadi, J. (2020). The promise of automated compliance checking. Developments in the Built Environment, 5, 100039. https://doi.org/10.1016/j.dibe.2020.100039

[4] Greenwood, D., Lockley, S. R., Malsane, S., & Matthews, J. (2010). Proceedings of the Construction, Building and Real Estate Research Conference of the Royal Institution of Chartered Surveyors held on 2-3 September 2010 in Paris, France. https://www.researchgate.net/publication/268186729_Automated_compliance_checking_using_building_information_models

[5] Hasan, M. M. (2016). Regulatory requirements Compliance in requirements engineering. International Journal of Systems and Service-Oriented Engineering, 6(4), 22–35. https://doi.org/10.4018/ijssoe.2016100102

[6] Weigand, H., Van Den Heuvel, W., & Hiel, M. (2011). Business policy compliance in service-oriented systems. Information Systems, 36(4), 791–807. https://doi.org/10.1016/j.is.2010.12.005

[7] Jureta, I. J., Siena, A., Mylopoulos, J., Perini, A., Susi, A., Fonds de la Recherche Scientifique – FNRS, University of Namur, FBK-Irst, University of Trento, FBK-Irst, & FBK-Irst. (2010). Theory of Regulatory Compliance for Requirements Engineering. Theory of Regulatory Compliance for Requirements Engineering. https://www.researchgate.net/publication/45902050_Theory_of_Regulatory_Compliance_for_Requirements_Engineering

[8] Ingolfo, S., Siena, A., Mylopoulos, J., Susi, A., & Perini, A. (2012). Arguing regulatory compliance of software requirements. Data & Knowledge Engineering, 87, 279–296. https://doi.org/10.1016/j.datak.2012.12.004

[9] Ingolfo, S., Siena, A., & Mylopoulos, J. (2011). Establishing Regulatory Compliance for Software Requirements. In Lecture notes in computer science (pp. 47–61). https://doi.org/10.1007/978-3-642-24606-7_5

[10] Shahin, M., Babar, M. A., & Zhu, L. (2017). Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices. IEEE Access, 5, 3909–3943. https://doi.org/10.1109/access.2017.2685629

[11] Gholami, A., & Laure, E. (2015). Security and Privacy of Sensitive Data in Cloud Computing : A Survey of Recent Developments. Security and Privacy of Sensitive Data in Cloud Computing : A Survey of Recent Developments, 131–150. https://doi.org/10.5121/csit.2015.51611

[12] Yimam, D., & Fernandez, E. B. (2016). A survey of compliance issues in cloud computing. Journal of Internet Services and Applications, 7(1). https://doi.org/10.1186/s13174-016-0046-8

[13] Rahmouni, H. B., Munir, K., Odeh, M., & McClatchey, R. (2012). Risk-Driven compliant access controls for clouds. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1202.5482

[14] Lie, M. F., Sánchez-Gordón, M., & Colomo-Palacios, R. (2020). DevOps in an ISO 13485 Regulated Environment: A Multivocal Literature Review. DevOps in an ISO 13485 Regulated Environment: A Multivocal Literature Review. https://arxiv.org/pdf/2007.11295

[15] Laukkarinen, T., Kuusinen, K., & Mikkonen, T. (2018). Regulated software meets DevOps. Information and Software Technology, 97, 176–178. https://doi.org/10.1016/j.infsof.2018.01.011

[16] García-Galán, J., Pasquale, L., Grispos, G., & Nuseibeh, B. (2016). Towards adaptive compliance. Towards Adaptive Compliance, 108–114. https://doi.org/10.1145/2897053.2897070

[17] Alkalbani, A., Deng, H., & Kam, B. (2016). Investigating the role of socio-organizational factors in the information security compliance in organizations. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1606.00875

[18] Massey, A. K., Smith, B., Otto, P. N., Antón, A. I., & North Carolina State University. (2011). Assessing the accuracy of legal implementation readiness decisions. In 2011 IEEE 19th International Requirements Engineering Conference Research Paper (p. 207) [Conference-proceeding]. IEEE. https://doi.org/10.1109/RE.2011.6051641

[19] Polu, O. R. (2021). AI-DRIVEN GOVERNANCE FOR MULTI-CLOUD COMPLIANCE: AN AUTOMATED AND SCALABLE FRAMEWORK. International Journal of Cloud Computing, 1(4), 1–13. https://doi.org/10.34218/ijcc_01_04_001

[20] Nekvi, M. R. I., & Madhavji, N. H. (2014). Impediments to regulatory compliance of requirements in contractual systems engineering projects. ACM Transactions on Management Information Systems, 5(3), 1–35. https://doi.org/10.1145/2629432