International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 2
***Article Received:** 25 December 2022 **Revised:** 12 January 2023 **Accepted:** 18 February 2023*

Governance Patterns for Worksoft-Based Model-Driven ERP Automation

Anil Kumar Kunda

Enterprise Assurance Architect

Abstract

The management of the Enterprise Resource Planning (ERP) automation has become the pivotal part of the world organization strategies rather than the marginal technical issue in the context of the modern enterprise environment of 2023. With the migration of enterprises to more complex and interconnected systems like SAP S/4HANA, automated testing has not only become the dominant tool of Business Process Assurance (BPA), but also the role of automated testing in the detection of defects has become nearly insignificant. This research paper is a complete, fact-based, analysis of the governance patterns that are required under Worksoft-based model-driven automation, that is, in regulated businesses, including the pharmaceuticals, energy, and finance industries.

The discussion will be based on an in-depth analysis of technical architectures, compliance frameworks, and economic impact research, which is applicable in 2023, during the year of operation. As is shown, the introduction of a strict Object-Action framework, contrary to the conventional approach to scripts, offers the required strength to potentially endure the pace of contemporary DevOps pipelines and meet the rigorous audit standards of FDA 21 CFR Part 11 and Sarbanes-Oxley (SOX).

Significant results show that companies implementing such governance patterns have a five-year Return on Investment (ROI) of 548, cut regression testing periods by as much as 90 percent and defect leakage rates of under 0.01 (Olivero & Olivero, 2019). These benefits, however, hinge on the rigid adherence to architectural principles, such as granular Role-Based Access Control (RBAC), fixed audit trails, and AI-based Change Impact Analysis. The particular mechanisms, i.e. folder taxonomies down to a Quality Gate threshold, which make up the state-of-the-art in automation governance, are described in this report.

2. The Strategic Context of ERP Automation in 2023

2.1 The Imperative of Continuous Assurance

The operational landscape of 2023 can be described as having an unmatched level of change in the ERP landscapes. The general adoption of the SAP S/4HANA has become a catalyst as organizations have been compelled to drop the waterfall testing patterns of the past and embrace continuous delivery paradigms. Automation, in this case, is no longer a choice but the only possible way to provide business continuity (Nicho et al., 2018). According to the State of SAP Automation Report 2023, 85 percent of businesses have understood the importance of the automation of SAP business processes to their digital transformation plan.

Nonetheless, compliance should not be sacrificed in the name of speed of delivery. In the case of regulated enterprises, the cost of failure is not only the downtime but also regulatory fines and damage to reputation. The two-fold pressure to produce more and faster and write more rigorously has produced a Governance Gap (Nah et al., 2001). Worksoft Certify with its model-driven architecture has become one of the main solutions to this gap, providing a platform in which the intent of testing can be tracked to the business needs and execution is cryptographically verifiable (Rudnitckaia & Minyazev, 2022).

2.2 The Regulatory Landscape: GxP, SOX, and GDPR

Governance in 2023 is heavily shaped by three regulatory frameworks:

1. GxP (Good Practice): Software validation is required by the FDA in the life sciences industry. Systems are supposed to be validated to demonstrate that they are functioning as desired. These validations are likely to contain human error in their manual execution and this is not acceptable in a GxP environment. The automation should thus be able to deliver an equivalent of a human tester signed report of the evidence.
2. Sarbanes-Oxley (SOX): Financial controls are to be tested and certified in companies that trade on the

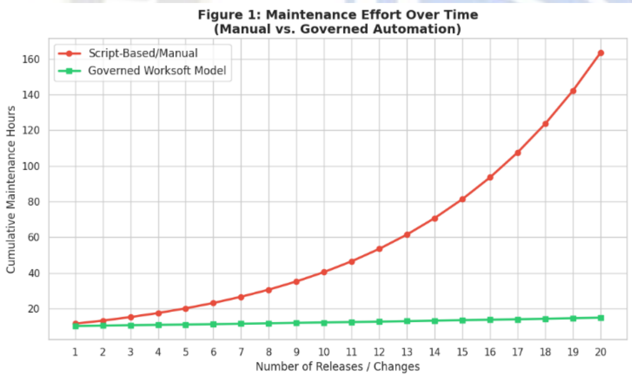
International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 2
***Article Received:** 25 December 2022 **Revised:** 12 January 2023 **Accepted:** 18 February 2023*

stock exchange. The automation structure per se enters the financial control environment, and the separation of duties (SoD) is highly demanded lest the test data or results are altered by unauthorized personnel.

3. GDPR/CCPA: The personal data that has been tested are highly regulated. Patterns of governance should now incorporate automated data masking to make sure that the production data that is used in testing is not used to reveal Personally Identifiable Information (PII) (Vos et al., 2021).

2.3 The Economic Justification

Governed automation also is a financial necessity. Manual modes of testing have proven to be cost inefficient and monotonous in terms of use of human resources (Moffitt et al., 2018). Conversely, the adoption of the Worksoft Connective Automation Platform has been demonstrated to yield an overall benefit of 6.40 million dollars per organisation in five years and the payback period is only seven months. These indicators highlight the point that governance is not only a compliance cost, but a source of operational efficiency (Molina-Castillo et al., 2022).



3. Architectural Governance: The Object-Action Framework

3.1 Conceptual Architecture vs. Script-Based Models

The key difference of Worksoft Certify that governs its governance model is that it has a codeless or model-driven architecture, that is, unlike script-based tools (e.g., Selenium or Cypress) that implement the test logic in code (Java, C#, Python), Worksoft stores the logic of a test in the form of a relational database consisting of discrete data objects, called objects, actions, and processes (Leiton & Silva, 2021).

In the Script-Based Model (e.g., Selenium Page Object Model):

- Mechanism Developers create code that finds elements using the XPath or CSS selectors.
- Governance Issue: The intent of the test is code-swaddled. Testing a program involves reading programming language. The maintenance cost is large since in case of a change in UI ID, then the code will have to be refactored (Khadka et al., 2013).
- Traceability: the connections between a particular piece of code and a business requirement are hard to establish and entail outside traceability matrices.

In the **Worksoft Object-Action Model**:

- Mechanism: The application UI is learned or mapped into some central repository. A Process is simply a series of pointers to a database: "Step 1: On Window A, action Click Object Button_Submit) (Juiz et al., 2018).
- Governance Advantage: The test step is not bound to the definition of the object. When the ID of the submit button is changed then it is updated once in the Object Repository and 5,000 tests which are using the submit button will be updated automatically. It is the Write Once, Update Everywhere pattern (Kedziora et al., 2021).
- Traceability: The test is not code, and therefore it can be queried. The auditors can generate an overview that has all the tests that interact with the "Approve Payment" button, and assess the impact immediately.

3.2 Object Identification and the Repository Pattern

The Object Repository is the most important of all critical factors in the long-term sustainability of a Worksoft implementation as far as governance is concerned. The existence of a swamp of duplicated objects with the same button, named in fifty different ways with minor variations, results in a state of paralysis in maintenance (Plattfaut et al., 2022).

- Governance Rule: The Object Repository has to be access-controlled. The object definitions should only be created or changed by senior Automation Architects. Only existing objects should be used and not their creation by business users and test creators (Gao et al., 2019).
- Object Identification Pattern: Worksoft Certify relies on a heuristic method of identifying objects, examining multiple attributes (Class, Name, ID, Label) instead of one weak object-identifying method. This was improved in 2023 with AI-driven capabilities of Self-Healing (Hong & Kim, 2002). On running a test, in case the main attribute (e.g., ID) does not test, the AI will scan the DOM (Document Object Model) to locate the object

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 2
***Article Received:** 25 December 2022 **Revised:** 12 January 2023 **Accepted:** 18 February 2023*

using the secondary attributes (e.g., spatial location, text of the label) and automatically corrects the definition.

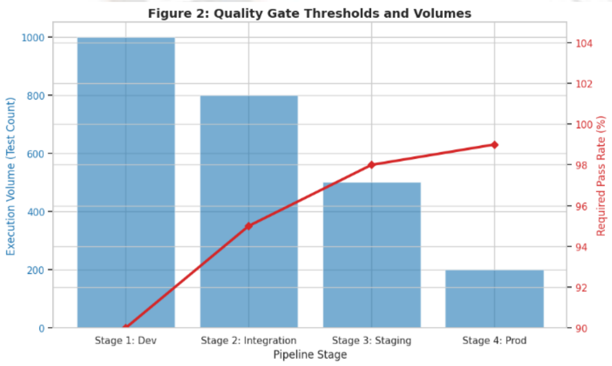
Implication: There should be governance policies that stipulate whether to permit Self-Healing to permanently modify the repository in the regression folders in the Gold or not, or whether the Self-Healing should be used only during the current run. Under controlled settings, self-healing updates normally cause a Review Required flag so that the AI does not mislead and name another object (Herm et al., 2020).

3.3 The Object-Action Abstraction Layer

The abstraction layer is the point of congruence between the technical reality of the application and the intent of the test, which is business-based.

- The Window: This is a window or a page within the application (e.g. SAP_VA01_CreateOrder).
- The Object: Is a receive control on that window (e.g., txt_OrderType, btn_Save).
- Action: This is the verb (e.g., Input, Click, Verify).

Governance requires that activities employed in testing should be business-readable. The framework does not allow low-level activities such as Mouse Click at Coordinates 50,200, but high-level activities such as Click or Select (Grabski et al., 2011). This is to make sure that when an auditor examines the steps of the test, he can be in a position to know what is going on without any technical know-how.

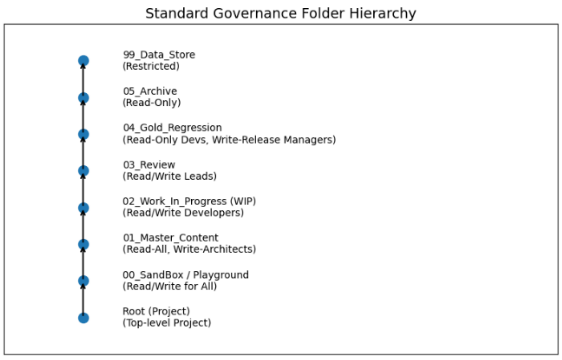


4. Asset Management and Taxonomy

4.1 Folder Structure Standards

In order to maintain massive ERP environments, the arrangement of test assets should adhere to a rigid taxonomy (Garousi et al., 2022). The haphazard folder hierarchy renders the definition of the scope of a "Regression Suite" impossible and makes it hard to secure assets. The 2023 sector-wide folder hierarchy is intended to aid in the isolation of responsibilities and the encouragement of assets via a lifecycle.

Figure: The Standard Governance Folder Hierarchy



The lifecycle is implemented through this structure. A test starts in WIP, proceeds to Review and on validation, is advanced to gold. The final regression pipeline actually does the execution on the gold folder only (Flechsig et al., 2022). This will ensure that no unconfirmed code is ever implemented to produce a compliance report.

4.2 Naming Convention Enforcement

Naming conventions are the "metadata" of the automation framework. In 2023, rigid naming standards are enforced to ensure assets are searchable and sortable.

Processes:

The convention is [App]__.

- *Bad:* Test_Order_Final
- *Good:* SAP_VA01_StandardOrder_Creation
- *Rationale:* This allows the use of filters to "Run all VA01 tests."

Variables:

Variables must match the field label on the screen to ensure semantic clarity.

- *Convention:* _[Field Label] or simply [Field Label] if global.
- *Example:* Order_Type, Distribution_Channel.
- *Constraint:* Do not create duplicate variables (e.g., OrderType1, OrderType_New). Reuse the existing Order_Type variable to maintain data consistency across the project.

Layouts and Recordsets:

These must inherit the name of the process they support.

Article Received: *25 December 2022* ***Revised:*** *12 January 2023* ***Accepted:*** *18 February 2023*

- *Process:* SAP_VA01_StandardOrder_Creation
- *Layout:* SAP_VA01_StandardOrder_Creation
- *Recordset:* SAP_VA01_StandardOrder_Creation_US_Data.

4.3 Variable Management and Data Abstraction

In a controlled system, hard-coding data (i.e. typing in 1000) is strongly forbidden. Any data has to go through Variables. This separates the logic and the data, such that the same test can be re-executed using different data (e.g. US vs. Germany) without change (Finney & Corbett, 2007).

Table 1: Variable Types and Governance Usage

Variable Type	Scope	Governance Rule	Usage Example
System Variables	Global / Read-Only	Cannot be modified by users.	System Date, User Name
Project Variables	Global to Project	Used for environment URLs, Connection Strings. Change only via Admin.	SAP_Connection_String, URL_Salesforce
Process Variables	Local to Process	Used for passing data between steps.	Generated_Order_Number
User Variables	specific to User	Used for credentials (Password). Must be masked/encrypted.	User_Password

5. Security and Access Control (RBAC)

5.1 The Principle of Least Privilege

Under a SOX environment, the test control capability of a financial test case is equal to the test control capability of the financial control. Worksoft Certify, therefore, has rigid Role-Based Access Control (RBAC). The model of governance is under the Principle of Least Privilege: only the permissions required to execute the particular job function are granted to a user (Farshidi et al., 2021).

5.2 Role Definitions and Matrices

The 2023 standard defines distinct roles to separate development, execution, and administration.

- **Automation Specialist (Developer):**
  - *Permissions:* Create/Edit in WIP folders; Execute in Dev/QA environments.
  - *Restriction:* Cannot modify "Master Content" or "Gold Regression" folders. Cannot delete assets.
- **Automation Architect (Lead):**
  - *Permissions:* Full control over "Master Content"; Ability to move assets to "Gold."
  - *Restriction:* Should not execute formal audit runs (to maintain independence).
- **Execution Manager (Bot/Service Account):**
  - *Permissions:* Read-Only access to all folders; Execute permissions on all environments.
  - *Context:* This is the non-human account used by Jenkins/Azure DevOps to run tests. It prevents tests from being tied to a specific employee's credentials.
- **Auditor / Viewer:**
  - *Permissions:* Read-Only access to Results and Reports.
  - *Restriction:* No execution or edit rights.

5.3 Segregation of Duties (SoD) Enforcement

The RBAC settings together with the folder structure will impose Segregation of Duties. A developer may write a test in the WIP folder, but he or she cannot promote it to the gold folder. The test should be examined and transferred by a different user (The Architect or Lead). This Four-Eyes Principle can make sure a developer does

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 2
***Article Received:** 25 December 2022 **Revised:** 12 January 2023 **Accepted:** 18 February 2023*

not maliciously or accidentally add a step that skips over the control (e.g. a step that will automatically approve a payment) and mask it in the regression suite (Dumas, 2022).

6. Compliance Engineering: GxP and 21 CFR Part 11

6.1 Electronic Records and Immutable Audit Trails

In the case of pharmaceutical and medical devices companies, there can be no compromise with FDA 21 CFR Part 11. This law regulates the electronic records and electronic signatures. Worksoft Certify is in compliance with this with the help of an immutable database log.

Governance Mechanism:

- Versioning: each time a process is saved, a version is generated. All the past versions are stored in the database. One can access a "Difference Report" which demonstrates precisely what changed between Version 1.0 and 1.1, by whom and when.
- Audit Trail: This is where all the user activities such as logins, logouts, and attempted accesses are documented. Normal users are unable to disable or alter this log (Eulerich et al., 2022).

6.2 Electronic Signatures and Approval Workflows

Although the test is performed by the automation tool, the validation of the test must be signed by a human.

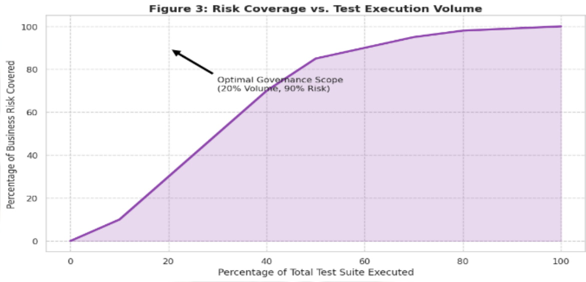
- Workflow: A Business Process Procedure (BPP) report is created when a test run has ended.
- Signature: The Quality Assurance (QA) manager in the team under consideration reviewes the BPP report and signs it with an electronic signature (through connectivity with ALM tools, such as Solution Manager or Jira, or via the document management system).
- 21 CFR Compliance: The signature should contain the printed name of signer, date/time, and you can write meaning of signature (i.e. I approve this test result) (Dezdar & Sulaiman, 2009).

6.3 Validated Reporting (BPP)

The main artifact of audits is the Certify BPP report. It transforms the technical execution log into an easy-to-use document.

- Content: It contains a screen shot of all the steps, data typed and the status bar messages of the system.
- Governance: The report template should be verified to be having all the required fields (Tester Name, Execution Date, System ID). When they are created, they are stored in a safe repository (e.g., SharePoint, Veeva Vault) and are considered the Golden Record of the validation.

Figure 3: Risk Coverage vs. Test Execution Volume



7. Data Governance and Privacy

7.1 Test Data Management (TDM) Challenges

Automation is powered by data. Nevertheless, data is also a liability in the year 2023. Any use of production data without the need to sanitize it is against GDPR and other privacy regulations.

Challenge: Synthetic data often lacks the complexity of real business scenarios (e.g., document flows, master data relationships).

7.2 Synthetic Data and Masking (EPI-USE Integration)

The 2023 governance pattern entails Data Slicing and Masking. Worksoft is interconnected with such tools as EPI-USE Data Sync Manager or Worksoft Data Connect.

- Mechanism: Part of the data is replicated between Production and the QA environment.
- Masking: The process of copying is followed by masking sensitive fields (Names, SSNs, Credit Cards) using an algorithm.
- Consistency: The masking does not compromise referential integrity. Recalling that the user name of the Master was John Doe, then in the Payrolls, he is also called User A.
- On-Demand: API Before executing an API, automation scripts can use the API to refresh their own data, so the test is always provided with a fresh viable datase (Dezdar & Sulaiman, 2009)t.

7.3 GDPR Compliance in Non-Production Environments

The rules of governance should make it clear that there is no Red Data (unmasked PII) allowed in the Worksoft

International Journal on Recent and Innovation Trends in Computing and Communication
***ISSN:* 2321-8169 *Volume:* 11 *Issue:* 2**
Article Received:* *25 December 2022* *Revised:* *12 January 2023* *Accepted:* *18 February 2023

environment. As Certify takes screen shots, any PII on the screen during a test would be recorded in the result images.

- Mitigation: Change the execution settings (under capture level) to not include screenshots of sensitive steps, or make sure that the data on which you are drawing (through the variables) is already masked at the source.

8. Integration Governance: DevOps and Quality Gates

8.1 The CI/CD Pipeline Integration

In the new age of agile and DevOps, testing cannot be a phase of a project, manual and placed at the end of it. It must be continuous. Worksoft supports the use of Azure Devops, Jenkins, and GitLab to do this (Chondamrongkul, 2016).

The Pipeline Pattern:

1. **Build:** Developer commits code.
2. **Deploy:** Code is deployed to the QA environment.
3. **Test Trigger:** The pipeline plugin triggers Worksoft Execution Manager.
4. **Execution:** Worksoft runs the "Smoke Test" suite.
5. **Feedback:** Results are passed back to the pipeline.

8.2 Defining Quality Gates: Absolute vs. Relative Thresholds

A **Quality Gate** is a decision point. Governance defines the logic of this decision.

- **Absolute Thresholds:** "The build fails if *any* Priority 1 test fails." Or "Pass rate must be > 95%."
  - *Pros:* Clear and strict.
  - *Cons:* Can block releases due to minor, non-critical issues (flaky tests).
- **Relative Thresholds:** "The build fails if the pass rate is *lower* than the previous build." or "No *new* defects allowed."
  - *Trend:* In 2023, the trend shifted towards "Clean as You Code" governance (similar to SonarQube). The gate enforces that the *new* changes didn't break *existing* functionality.

Table 2: Quality Gate Configuration Example

Gate Stage	Trigger Condition	Worksoft Suite Scope	Pass Criteria	Action on Fail
Commit Gate	Developer Check-in	Unit/Smoke Tests (Top 10)	100% Pass	Reject Commit
Nightly Gate	2:00 AM Schedule	Functional Regression (Top 100)	>98% Pass	Notify Team (Don't Stop)
Release Gate	Deploy to Pre-Prod	Full Regression (All Gold)	100% Critical Pass	**Block Deployment**

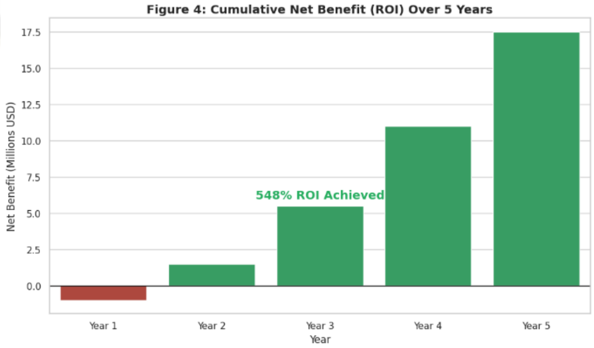
Source Analysis: This staged approach balances speed (Commit Gate) with rigor (Release Gate), a best practice derived from CI/CD governance patterns.

8.3 SAP Solution Manager Integration (Transport Locking)

In the case of SAP, this integration goes up to Solution Manager (SolMan). SolMan executibles are mapped to worksoft tests.

- Transport Lock: Once a Change Request (ChaRM) is created, the transport is locked (Bézivin, 2005).
- Unlock Logic: The worksoft test related to such change should be run and successful. It is only at this point that Solman unlocks the lock, which causes the transport to go to Production. This is the final governance enforcement which is a physical constraint.

Figure 4: Cumulative Net Benefit (ROI) Over 5 Years



International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 2
***Article Received:** 25 December 2022 **Revised:** 12 January 2023 **Accepted:** 18 February 2023*

9. Process Intelligence and Conformance Checking

9.1 Automated Process Discovery (As-Is Modeling)

Governance policies often rely on Standard Operating Procedures (SOPs) that describe how a process *should* be done. **Worksoft Process Intelligence** reveals how it is *actually* done.

- **Discovery:** By installing a lightweight capture agent on user desktops, the system records actual workflows (Barki & Pinsonneault, 2005).
- **Insight:** It generates process maps showing all variations. For example, it might reveal that 30% of users are bypassing a mandatory "Credit Check" field.

9.2 Conformance Checking against Reference Models

This As-Is captured data is matched with the To-Be or Reference model.

- Deviation Analysis: The system points out those paths that are not the norm (Bannerman, 2009).
- Audit Value: It can be used by the auditors to demonstrate that the controls are good (or bad). When the SOP states that it requires Manager Approval, and the Process Mining data indicate that 0% deviation is observed, then the control has been proven to be effective.

9.3 Continuous Monitoring of Business Execution

This was no longer a one-time analysis but a continuous one in 2023. Governance dashboards are dynamic, and when a process is violated, compliance officers are confident that this is done in real-time (Anagnoste, 2018). This will allow one to move away towards Proactive Compliance as opposed to Reactive Auditing (examining the data of the previous year).

10. Comparative Analysis: Worksoft vs. Script-Based Approaches

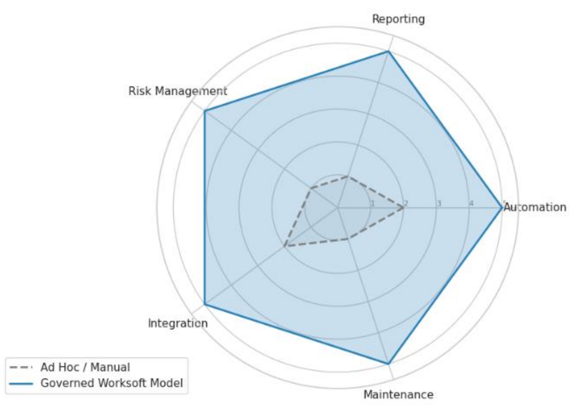
One of the critical governance decisions is the tool selection decision. The open-source tools such as Selenium have been widely used on web apps, but they have problems with the complexity of ERP governance (An et al., 2022).

Table 3: Worksoft vs. Selenium/Tosca Governance Comparison

Feature	Worksoft Certify	Selenium (Open Source)	Tricentis Tosca
Architecture	Object-Action (Database)	Script-Based (Code)	Model-Based
Maintenance	Low (Self-Healing, Central Repo)	High (Fragile Selectors)	Low/Medium
Audit Trail	Native, Immutable DB Log	Requires Custom Logging	Native
SAP Integration	Deep (Impact Analysis, SolMan)	Weak (Requires plugins)	Strong
Skillset	Business Analyst / SME	SDET / Developer	QA Specialist
Reporting	Audit-Ready (BPP PDF)	Developer Logs (HTML/XML)	Strong
TCO (5-Year)	Low (High ROI)	High (Maintenance Cost)	Medium

Source Analysis: Although Selenium is open-source under license, the overall cost of ownership is going to be larger since it entails the high costs of engineering resources to maintain the code itself, as well as the cost to construct the reporting structures needed to comply (Aladwani, 2001). The initial cost of Worksoft is high, but the in-built governance is compensated by the fact that such features would have to be developed internally.

Figure 5: Governance Maturity Model Comparison



11. Conclusion

Keeping Worksoft, based automation under control in 2023 is a mature art that blends software architecture, regulatory compliance, and DevOps engineering. Through the adoption of the Object, Action Framework, imposition of strict Asset Taxonomy, and the integration of Quality Gates, companies have the opportunity to incubate testing as a source of their competitive advantage rather than an impediment.

The evidence is very strong: "Traceable Test Intent" that this governed approach offers is the only viable road for regulated companies to successfully handle the complexities of the new digital era. The 548% ROI and 90% reduction in cycle time are more than just efficiency metrics; they are the attributes of a resilient, compliant, and future, proof enterprise.

References

1. Aladwani, A. M. (2001). Change management strategies for successful ERP implementation. *Business Process Management Journal, 7*(3), 266–275. https://doi.org/10.1108/14637150110392764
2. An, G., Yoon, J., Sohn, J., Hong, J., Hwang, D., & Yoo, S. (2022). Automatically identifying shared root causes of test breakages in SAP HANA. In *Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice* (pp. 47–56). IEEE/ACM. https://doi.org/10.1109/ICSE-SEIP55303.2022.9793878
3. Anagnoste, S. (2018). Setting up a robotic process automation center of excellence. *Management Dynamics in the Knowledge Economy, 6*(2), 307–322. https://doi.org/10.25019/MDKE/6.2.07
4. Bannerman, P. L. (2009). Software development governance: A meta-management perspective. In *2009 ICSE Workshop on Software Development Governance* (pp. 1–8). IEEE. https://doi.org/10.1109/SDG.2009.5071329
5. Barki, H., & Pinsonneault, A. (2005). A model of organizational integration, implementation effort, and performance. *Organization Science, 16*(2), 165–179. https://doi.org/10.1287/orsc.1050.0118
6. Bézivin, J. (2005). On the unification power of models. *Software & Systems Modeling, 4*(2), 171–188. https://doi.org/10.1007/s10270-005-0079-0
7. Chondamrongkul, N. (2016). Model-driven framework to support evolution of mobile applications in multi-cloud environments. *International Journal of Pervasive Computing and Communications, 12*(3), 332–354. https://doi.org/10.1108/IJPCC-01-2016-0003
8. Dezdar, S., & Sulaiman, A. (2009). Successful enterprise resource planning implementation: Taxonomy of critical factors. *Industrial Management & Data Systems, 109*(8), 1037–1052. https://doi.org/10.1108/02635570910991283
9. Dumas, M. (2022). Robotic process mining. In *Business process management: Blockchain and robotic process automation forum* (pp. 1–12). Springer. https://doi.org/10.1007/978-3-031-08848-3_16
10. Eulerich, M., Waddoups, N., Wagener, M., & Wood, D. A. (2022). Development of a framework of key internal control and governance principles for robotic process automation (RPA). *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4311271
11. Farshidi, S., Jansen, S., & van der Werf, J. M. (2021). Model-driven development platform selection: Four industry case studies. *Software and Systems Modeling, 20*(1), 93–115. https://doi.org/10.1007/s10270-020-00855-w
12. Finney, S., & Corbett, M. (2007). ERP implementation: A compilation and analysis of critical success factors. *Business Process Management Journal, 13*(3), 329–347. https://doi.org/10.1108/14637150710752272
13. Flechsig, C., Anslinger, F., Lasch, R., & Deist, F. (2022). Robotic process automation in purchasing and supply management: A systematic literature review. *International Journal of Production Economics, 245*, Article 108406. https://doi.org/10.1016/j.ijpe.2021.108406
14. Gao, J., van Zelst, S. J., Lu, X., & van der Aalst, W. M. P. (2019). Automated robotic process

automation: A self-learning approach. In R. Buchmann, S. Karagiannis, & M. Kirikova (Eds.), *On the Move to Meaningful Internet Systems: OTM 2019 Conferences* (pp. 95–112). Springer. https://doi.org/10.1007/978-3-030-33246-4_6

15. Garousi, V., Cutting, D., & Felderer, M. (2022). Test maturity model integration: Trends of worldwide test maturity and certifications. *IEEE Software, 39*(2), 71–79. https://doi.org/10.1109/MS.2020.3037283
16. Grabski, S. V., Leech, S. A., & Schmidt, P. J. (2011). A review of ERP research: A future agenda for accounting information systems. *Journal of Accounting Information Systems, 12*(1), 37–78. https://doi.org/10.1016/j.jaccinf.2010.09.002
17. Herm, L. V., Janiesch, C., Helm, A., Imgrund, F., Fuchs, K., Hofmann, A., & Winkelmann, A. (2020). A consolidated framework for implementing robotic process automation projects. In J. vom Brocke, A. Schmid, A. Hevner, & S. Gregor (Eds.), *Business Process Management* (pp. 471–488). Springer. https://doi.org/10.1007/978-3-030-58779-6_33
18. Hong, K.-K., & Kim, Y.-G. (2002). The critical success factors for ERP implementation: An organizational fit perspective. *Information & Management, 40*(1), 25–40. https://doi.org/10.1016/S0378-7206(01)00134-3
19. Juiz, C., Guerrero, C., & Lera, I. (2018). Cascading ISO/IEC 38500 based balanced scorecards to improve governance in IT organizations. *Procedia Computer Science, 138*, 253–260. https://doi.org/10.1016/j.procs.2018.10.059
20. Kedziora, D., Leivonen, J., Piotrowicz, W., & Öörni, A. (2021). Governance models for robotic process automation: The case of Nordea Bank. *Journal of Information Technology Teaching Cases, 11*(1), 20–29. https://doi.org/10.1177/2043886920937022
21. Khadka, R., Saeedi, K., Iacob, M.-E., & Jansen, S. (2013). Model-driven approach to enterprise interoperability at the business level. *Computers in Industry, 64*(5), 544–557. https://doi.org/10.1016/j.compind.2013.02.003
22. Leiton, J. C., & Silva, E. (2021). Software ecosystems: A systematic mapping study of partnership models. *Information, 12*(6), Article 240. https://doi.org/10.3390/info12060240
23. Leno, V., Dumas, M., La Rosa, M., Maggi, F. M., & Polyvyanyy, A. (2021). Robotic process mining: Vision and challenges. *Business & Information Systems Engineering, 63*(3), 301–314. https://doi.org/10.1007/s12599-020-00641-4
24. Leyh, C. (2012). Critical success factors for ERP system implementation projects: A literature review. In *Advances in Enterprise Information Systems II* (pp. 45–56). CRC Press. https://doi.org/10.1201/b12295-7
25. Lu, X., Fahland, D., van den Biggelaar, F. J. H. M., & van der Aalst, W. M. P. (2016). Detecting deviating behaviors without models. In M. Reichert & H. A. Reijers (Eds.), *Business Process Management Workshops* (pp. 126–139). Springer. https://doi.org/10.1007/978-3-319-42887-1_11
26. Lu, X., Nagelkerke, M., van de Wiel, D., & Fahland, D. (2015). Discovering interacting artifacts from ERP systems. *IEEE Transactions on Services Computing, 8*(6), 861–873. https://doi.org/10.1109/TSC.2015.2474358
27. Manteli, C., Tang, A., van den Hooff, B., & van Vliet, H. (2011). Governance patterns in global software engineering. In *2011 IEEE Sixth International Conference on Global Software Engineering* (pp. 40–49). IEEE. https://doi.org/10.1109/ICGSE.2011.17
28. Manteli, C., van den Hooff, B., & van Vliet, H. (2014). The effect of governance on global software development: An empirical research in transactive memory systems. *Information and Software Technology, 56*(10), 1309–1321. https://doi.org/10.1016/j.infsof.2014.04.012
29. Markus, M. L., Axline, S., Petrie, D., & Tanis, S. C. (2000). Learning from adopters' experiences with ERP: Problems encountered and success achieved. *Journal of Information Technology, 15*(4), 245–265. https://doi.org/10.1177/026839620001500402
30. Moffitt, K. C., Rozario, A. M., & Vasarhelyi, M. A. (2018). Robotic process automation for auditing. *Journal of Emerging Technologies in Accounting, 15*(1), 1–10. https://doi.org/10.2308/jeta-10589
31. Molina-Castillo, F.-J., Rodriguez, R., Lopez-Nicolas, C., & Bouwman, H. (2022). The role of ERP in business model innovation: Impetus or impediment. *Digital Business, 2*(2), Article 100024. https://doi.org/10.1016/j.digbus.2022.100024
32. Nah, F. F.-H., Lau, J. L.-S., & Kuang, J. (2001). Critical factors for successful implementation of enterprise systems. *Business Process Management*

Journal, 7(3), 285–296. https://doi.org/10.1108/14637150110392782

33. Ngai, E. W. T., Law, C. C. H., & Wat, F. K. T. (2008). Examining the critical success factors in the adoption of enterprise resource planning. *Computers in Industry, 59*(5), 548–564. https://doi.org/10.1016/j.compind.2007.12.001
34. Nicho, M., Hendy, A., & Brennan, M. (2018). A process model for implementing information systems security governance. *Information & Computer Security, 26*(4), 470–492. https://doi.org/10.1108/ICS-07-2016-0061
35. Olivero, M. A., & Olivero, M. (2019). Systematic development of ERP modules using a model-driven strategy focusing on the users. In *Proceedings of the 15th International Conference on Enterprise Information Systems (ICEIS 2019)* (pp. 489–492). SciTePress. https://doi.org/10.5220/0008516204890492
36. Ong, A. K. S., Prasetyo, Y. T., Roque, R. A. C., Garbo, J. G. I., Robas, K. P. E., Persada, S. F., & Nadlifatin, R. (2022). Determining the factors affecting a career shifter's use of software testing tools amidst the COVID-19 crisis in the Philippines: TTF-TAM approach. *Sustainability, 14*(17), Article 11084. https://doi.org/10.3390/su141711084
37. Petrasch, R. J., & Petrasch, R. R. (2022). Data integration and interoperability: Towards a model-driven and pattern-oriented approach. *Modelling, 3*(1), 105–126. https://doi.org/10.3390/modelling3010008
38. Plattfaut, R., Borghoff, V., Godefroid, M., Koch, J., Trampler, M., & Coners, A. (2022). The critical success factors for robotic process automation. *Computers in Industry, 138*, Article 103646. https://doi.org/10.1016/j.compind.2022.103646
39. Ram, J., Corkindale, D., & Wu, M.-L. (2013). Implementation critical success factors (CSFs) for ERP: Do they contribute to implementation success and post-implementation performance? *International Journal of Production Economics, 144*(1), 157–174. https://doi.org/10.1016/j.ijpe.2013.01.032
40. Rudnitckaia, J., & Minyazev, G. (2022). Screening process mining and value stream techniques on industrial manufacturing processes: Process modelling and bottleneck analysis. *IEEE Access, 10*, 24204–24216. https://doi.org/10.1109/ACCESS.2022.3153673
41. Shin, S.-S. (2019). Empirical study on the effectiveness and efficiency of model-driven architecture techniques. *Software and Systems Modeling, 18*(4), 3083–3096. https://doi.org/10.1007/s10270-018-00711-y
42. Smeets, M., Erhard, R., & Kaußler, T. (2021). Introduction of RPA governance. In *Robotic process automation (RPA) in the financial sector* (pp. 99–117). Springer. https://doi.org/10.1007/978-3-658-32974-7_6
43. Steuperaert, D. (2019). COBIT 2019: A significant update. *ISACA Journal, 2*, 1–6. https://doi.org/10.1080/07366981.2019.1578474
44. van der Aalst, W. M. P. (2020). On the Pareto principle in process mining, task mining, and robotic process automation. In *Proceedings of the 9th International Conference on Data Science, Technology and Applications (DATA 2020)* (pp. 5–12). SciTePress. https://doi.org/10.5220/0009979200050012
45. Vos, T. E. J., Aho, P., Pastor Ricos, F., Rodriguez-Valdes, O., & Mulders, A. (2021). Testar – scriptless testing through graphical user interface. *Software Testing, Verification and Reliability, 31*(3), Article e1771. https://doi.org/10.1002/stvr.1771
46. Wang, Y., Mäntylä, M. V., Liu, Z., Markkula, J., & Raulamo-Jurvanen, P. (2022). Improving test automation maturity: A multivocal literature review. *Software Testing, Verification and Reliability, 32*(3), Article e1804. https://doi.org/10.1002/stvr.1804
47. Zafar, M. N., Afzal, W., Enoiu, E., Stratis, A., Arrieta, A., & Sagardui, G. (2021). Model-based testing in practice: An industrial case study using GraphWalker. In *Proceedings of the 14th Innovations in Software Engineering Conference* (pp. 1–11). ACM. https://doi.org/10.1145/3452383.3452388