INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.625

Harnessing Artificial Intelligence for Salesforce Metadata: Advanced Migration Strategies and Strategic Business Benefits

Bijal Lalitkumar Dave

Full Stack Lead, Istream Solution, USA

ABSTRACT: Artificial intelligence is an expanding speciality of computer science concentrated on developing state-of-the-art structures and algorithms to automate complex tasks. Salesforce configurations are becoming more and more complex; therefore, the demand for a rapid and accurate impact analysis increases. This supports the contention that manual methods for identifying changes cannot provide the advantages mentioned earlier since the manual impact analysis takes time and is prone to manual errors. Artificial intelligence can be applied to automate the process of identifying changed data in Salesforce. Thus, the use of artificial intelligence can increase the capability for quicker, accurate impact analyses while decreasing the amount of manual effort typically needed for both ease of audits and compliance. Artificial intelligence also allows for analysis of vast historical datasets, enabling a more accurate identification of changes while reducing false-positives. A significant advantage of using artificial intelligence is enhanced optimization of regression testing using intelligent dependency mapping, leading to consistent impact analyses using artificial intelligence, while enabling a quality check of the impact analyses with domain expertise. Quicker releases with lower operational costs can result. Artificial intelligence also provides robust solutions that are always learning based on the input of the user, although for organizational operational to be met, data intensive enterprise-specific models need often to be built. Consequently, AI systems related to Salesforce configurations are advancing more and more to provide sophisticated and efficient impact analysis and automation capabilities.

KEYWORDS: Artificial Intelligence, Salesforce, Regression Testing, Data-Intensive Models

I. INTRODUCTION

Salesforce metadata plays a vital role in defining the fabric and behavior of the Salesforce platform. Metadata is the collection of schema components which includes custom objects, fields, validation rules, page layouts, workflow automation, permission settings, and user interface components that actually displays and controls data in an organization's instance of Salesforce. It is this metadata that allows administrators and developers to customize Salesforce for specific business processes. Along with flexibility and scalability, metadata also has other benefits that cannot be accomplished with other alternatives (in comparison to all configuration data hard-coded). Configuration deployment, versioning, and migration between environments are also straightforward. Metadata also enables Salesforce AI capabilities to leverage context and structure to provide intelligent automation and improved user experiences. [1].

It is important to review multiple sets of metadata before the migration to Salesforce, to help ensure a successful transition post-migration. Custom Objects and Fields are the foundation of the data model. Reviewing Custom Objects and Fields helps to audit against data loss and allow for proper data configuration. Page Layouts will need to be audited for consistency and user satisfaction. Record Types are used to customize a UI display and drive certain business processes; auditing Record Types prevents potential conflicts with custom operations and protects from losing custom views. Data quality can be limited by the business with the use of validation rules, and auditing data quality ensures that business logic will be preserved after migration. Workflows and/or automation rules should be audited against prior dependencies so that they remain functional after the migration. Dashboards and reports are important for analytics, and auditing helps to ensure users will have access to analytics tools they will need; missing reports and dashboards can directly impact productivity. Apex Classes and Triggers provide the ability to customize standard capabilities or provide integrations with outside systems, and should be audited to help ensure completeness and ability to work upon migration. Custom user interface design, such as Visualforce Pages and Lightning Components for example, would need to be audited, to help ensure anything will work in the same way, as originally intended. Email

templates will need to maintain branding standard, and static resources such as images and/or scripts should be functional after migration. Finally, Custom Metadata Types and Custom Settings provide flexible behavior for an application; auditing should be completed to avoid misconfiguration of these types. In conclusion, a comprehensive auditing of migrated changes validates the application.

By taking advantage of Salesforce's AI capabilities, we will better prepare the CRM as an intelligent, agile platform that can allow your business to better utilize information for decision making, digital transformation, automation of repetitive tasks, and larger customer engagement with hyper-personalized experiences. AI allows Salesforce to enable unprecedented capabilities such as data mining, analyzing large amounts of data associated with customer behavior patterns, and offering suggestive insights for improved business performance through examples like recommendations, natural language processing, and Einstein Analytics. By leveraging AI, Salesforce can analyze data structures and logically scaled complex processes involved in marketing, sales, and service engagements with very limited human involvement or oversight. In addition, AI utilized the Salesforce metadata that allows the platform to comprehend data structures and their logical meanings, which allows the platform to scale complex processes without human oversight, while continuing to react to the changing ecosystem lifecycle business dynamics. This allows the engagement with customers to be executed consistently and governed with this Salesforce manner while scaling the capabilities to prepare for the next digital transformation, operational efficiency, innovation speed, and customer engagement for hyper-personalized experiences. In the end, AI allows your organization to operate more efficiently, agile, and to compete through its data and its metadata.

Currently, the use of AI solution for quality assurance and impact analysis is very popular in Salesforce. Rapid and flexible testing methodology and changing business needs has made the exploration of these tools become a reality as they facilitate automatation of the cycle of regular testing while achieving significant savings in operational costs. In agile development, AI can read user stories, offer suggestions for acceptance criteria, and conduct comprehensive testing all at the same time. Companies such as Opkey provide capability to dig deep with inventory and risk assessment to ascertain impact before changes are deployed, further minimizing issues and accentuating successful releases with greater assurance. AI can also identify issues, recommend tests and manage the Quality Assurance cycle all on its own when integrated into a continuous integration/continuous delivery ('CICD') pipeline, creating the additional capability for continuous testing, with the ability to observe in real-time. To successfully harness AI for Salesforce QA, organizations must seek to execute pilot projects within high value, leverage the capability of the QA professional, ensure accuracy of data, find the balance between human intelligence and AI automation, and refine the model iteratively based on outcomes. There are true and real QA challenges with Salesforce given its complexity and inherent customization, especially with regards to scaling and regression testing, and cadence of releases, but AI can make the problems more palatable by creating less risky and cheaper, safe releases more often.

The article discusses practical ways of sending Salesforce data while detailing the business benefits. It is designed to help organizations tackle data migration challenges while providing best practices, resources and approaches to migration from one Salesforce environment to another in a smooth, low-risk way. The systematic approach to migrating metadata involves several steps in the process, including planning, auditing, mapping dependencies, and using automation tools that alleviate the burden of supporting the customization, demonstrate no loss of data, and provide the organization with a clear business purpose in supporting the revamped metadata. This article lays out the strategic economic benefits of AI as a migration driver including: deploying quickly, better governance, less re-work, supporting the plans for agility, while also supporting and enabling agile delivery pipelines. AI-driven migration approaches support and preserve metadata integrity, while also providing the service of predictive impact analysis for accelerated change, where it is possible.

There are several considerations to take when migrating Salesforce metadata - project scope and goals for avoiding missed timelines and reworking; data quality for avoiding duplication and inconsistency; data mapping for avoiding operational failure and automation failure; prescriptive testing for avoiding problems until after migration; running automations after migration would have to be disabled, so as to avoid unintentional changes to data; testing the backup and rollback processes for data and uptime; awareness and limitations for performance, as this may potentially cause failure; awareness and communication for all stakeholders for managing expectations and reducing any miscommunications. After migration, there needs to be a way to track or look for any latent issues. Additionally, understanding the impact Salesforce metadata have for business processes to reduce disruption must not be overlooked.

To mitigate these risks, organizations should clearly define scope, cleanse, validate, and deduplicate data prior to migration, collaboratively plan data migration, test the migration thoroughly, disable automations, backup data and process, performance monitor, communicate transparently, and validate data process are the center post-migration. Preventative risk identification and mitigation are meant to (1) secure the integrity of data, (2) assist for continuity of business, and (3) provide a smooth hand-off process.

II. THE ROLE OF METADATA IN SALESFORCE AI

Salesforce metadata defines the structure, customization, and configuration of a Salesforce environment. Metadata is distinct from business data, which consists of real records such as accounts and contacts. The important attributes of Salesforce metadata include custom objects and fields, which are part of the data model; page layouts, which improve human consumption; record types for different business process; validation rules to maintain consistency in data; and workflows to automate tasks. Metadata is represented through dashboards and reports that provide insights; Apex classes and triggers that enhance further automation of tasks. Visualforce pages and Lightning components offer custom user interfaces, while other metadata features such as custom settings and email templates offer dynamic configurations. Salesforce's metadata is essential for flexibility and modularity in creating custom experiences to users. Moreover, metadata is a key principle in running and managing Salesforce environments as effectively as possible.

When migrating to Salesforce, certain metadata types have to be handled carefully to avoid compromising data and functional integrity. Custom objects and fields will also need to be migrated in order to keep some relational structure. Also, Apex classes and triggers will need to be thoroughly tested to ensure business logic is appropriately functioning. Automations such as workflows, process builders, and flows will also need to be tested due to dependencies. Security settings would also include the fact that sharing rules, permission sets, and profiles are maintained in order to guarantee users have proper access. The custom metadata types and settings should be re-created properly to avoid functional challenges. Record types and page layouts need to match perfectly so users have the same experience as they did prior to the migration. Reports and dashboards will need to carefully map the data source to avoid disruption in analytics. Visualforce pages and Lightning components will need to thoroughly tested to ensure compatibility post-migration. We will also need to properly migrate static resources and email templates to maintain consistency in communications and branding. Overall, due to all these interdependencies and complexity, it will require careful sequencing, testing, and validation to minimize disruption and ensure integrity in Salesforce.

Metadata provides critical insights into the data structure, data relationships, business rules, and system configuration that the AI needs to know and act upon. It is somewhat of a blueprint of how data is organized, the relationship between different objects, and the business rules that drive them. It allows the AI to provide flexible and scalable automation without being hooked to hard-coded logic. Metadata allows the AI to understand business objects and fields, recognize automation elements, personalize recommendations and interactions, and be responsive to changes in setups or schema changes. In summary, metadata enables the AI to deliver higher levels of productivity, to act accurately on data, and improve the user experience in Salesforce environments, transforming the AI's functionality from basic static automation to proactive and intelligent support based on business needs.

The metadata supplies essential foundation of abstraction that separates AI logic from actual specifics of company data and processes so that Salesforce AI Solutions can be adaptable. Metadata incorporates important entities such as objects, fields, relationships, and system behaviors, then contextualizes the Salesforce environment for AI agents without the need for hardwiring any assumptions. This adaptability allows AI to evolve with changing business requirements while requiring no code. Additionally, metadata provides consistency, which becomes especially imperative because once metadata becomes structured, everything creates a determined and predictable standardization to conduct work. In performing work, metadata standardizes language and structure, thereby ensuring that differing AI processes through differing AI instances can still communicate, and always have relatively the same functionality, and perplexity through the determination of relations, despite the plethora of data, or even automated workflows to address. This makes the AI both accurate and timely through group order of types and data continuity. This groups the AI for either predictive analytics, smart automation, or personalized user experience. Lastly, framework that are metadata enabled create operational collaboration capabilities that are real time and agile in hying improvements that reduces operational risk while improving good governance. Metadata ensures that Salesforce is dynamic and intelligent to put AI to work to efficiently help automate, optimize, and innovate business processes.[8]

Moving complex metadata in Salesforce poses quite a number of difficulties that take a considerable extent of care to remediate. Metadata has a number of dependencies since one item is reliant on another, so migrating an item out of order may create a complicated error. You must maintain the integrity of the data and correct mapping without creating data inconsistencies or loss. You may also need to be skilled in tooling, and you may need to generate the advanced configuration because of limitations on normal tooling, such as change sets. Ensuring versions stay the same within the development, testing, and production states takes a robust, disciplined process in conjunction with a version control system to maintain data synchronization. When you look to migrate automation or business logic, you will need to sequence them while also create an extensive test suite to validate that migration took place accurately or in the way you intended, along with strong rollback processes if the deployment fails. Furthermore, you will want to do all testing against your data in various levels while preserving validation for a successful migrate with the least impact of delay. Coordinating with the organization through its stakeholder members will help reduce misunderstandings; if a party does not understand then quality transitions will not happen. Finally, you will also need to set up separate sandboxes or test environments to see what the impacts will be of migrations; this essentially encourages extra work. In conclusion, thoughtful planning, acknowledgment of dependencies, an idea of some automated tools, thorough validation, and go-forward processes are essential for restricting risk while enabling seamless synchronization in complex Salesforce metadata migrations [9].

Several approaches and tools are recommended to evaluate and automate Salesforce metadata migration. Salesforce's own SFDX, a modern command-line interface, retrieves, deploys, and validates information across organizations, easily integrates with the CI/CD pipeline for automated migrations. AutoRABIT is an all-inclusive release management system featuring smooth dependency management, selective deployments, metadata transfers, rollback options, and detailed logging to help minimize errors. It serves as a very reliable alternative to ANT Migration Tool, which was suitable for more complex migrations. The new Salesforce CLI has replaced ANT Migration Tool in the course of this time. Changesets are generally a quick native solution for deployments between linked organizations. Nonetheless, there are limitations, including a lack of support for destructive changes and difficulty in managing large-scale migrations. Version control-Git-integrates the management of different environments and collaborative development. Metadata analysis, such as AutoRABIT's Org Sync, helps check the differences from source to destination organizations for the identification of discrepancies and further helps plan the migration. ETL and data migration-MuleSoft and Talend-can help with related data migration needs. In sum, proper planning of the migration scope, auditing of metadata, automation of retrieval and deployment, sandbox testing, and continuous monitoring and logging of migration activities are key recommendations. Overall, automating Salesforce environment management increases agility, speeds up the release process, and decreases the possibility of manual errors, which is represented in Table 1 below.

Table 1: Approaches and Tools for Assessing and Automating Salesforce Metadata Migration

Tool Name	Description	Best For	Key Features	Pros	Cons
Salesforce CLI (SFDX)	Command-line tool for retrieving and deploying metadata	Developers and DevOps teams	Scriptable, integrates with CI/CD, supports modular deployments	Modern, flexible, widely supported	Steeper learning curve for beginners
AutoRABIT	End-to-end release management platform	Enterprises and large orgs	Dependency management, rollback, reporting	Comprehensive automation	Paid licensing
ANT Migration Tool	Java-based CLI tool for scripted metadata deployment	Technical experts	Full control over deployment, scripting support	Free, highly customizable	Less user-friendly, legacy tool
Changesets	Salesforce native point-and-click deployment tool	Simple, connected org deployments	Easy UI, metadata selection	No setup required, built-in	Limited capability for complex/large migrations

Git & Version Control	Source control for metadata and code management	Agile and collaborative teams	Branching, version history	Enables collaboration and tracking	Requires integrated CI/CD setup
Metadata Analyzer Tools	Tools to compare org metadata differences and dependencies	Migration scoping and audit	Visual comparison, dependency mapping	Helps plan accurate migrations	May have cost or complexity limitations
MuleSoft / Talend	ETL and data integration platforms	Complex data and metadata migrations	Data transformation, scheduling	Robust for complex integrations	Learning curve, costs
IsyncSF	Native Salesforce app for org-to-org migration	Salesforce to Salesforce migrations	Data cleansing, masking, migration templates	Native integration and compliance	Paid, focused on Salesforce sources
Data Loader / DataLoader.io	GUI and CLI tools for bulk data import/export	Data migration, bulk operations	CSV support, scheduling, error logging	Free or freemium options	Primarily data-focused, limited metadata support
Skyvia	Cloud-based ETL and data migration	Small to medium business migrations	Visual UI, data sync, replication	Free tier available, easy to use	Feature limits on free plan

III. SYSTEM OVERVIEW

In order to satisfy Testing Salesforce Smart Works 2.0, the Robo project has built a solid, reliable, and scalable architecture focused on quality assurance, with a full complement of testing types from functional, integration, regression, performance, and security testing, focusing on automation and smart tools for agile development integration of testing. The key features are automation of change impacting testing with automated change detection, which highlights impacted components in ftalese metadata and processes to avoid missed dependencies in terms of pull requests; Dependency Mapping helps to convey the dependencies visually so that testing can be prioritized on business-critical processes; Smart Regression and Test Impact Analysis help with focused regression tests to speed up validation of tests and any only relevant tests are run. Guided dashboards and analytics support real-time decision-making by presenting impact analyses and scoping for testing. In addition to the above, the architecture facilitates compliance and transparency with logging and sharing of impact analysis feature and activity logs for regulated industries. The architecture also includes coordination of multiple testing teams; automation of regression testing; management of user acceptance testing; client walkthroughs; impact assessments of business processes; planning and monitoring of test cycles; support for integrations; and API validations and testing. This architecture is designed to further an iterative approach for agile teams to establish a cultural norm of 'continuous quality', effective coverage, and risk mitigation, accountability, quality peer-reviewed reflective practice, and guide consistency.

- **Automated Change Detection:** It tracks changes to Salesforce data, processes, and metadata and evaluates dependent components to avoid loss of dependencies and uncover more testing needs much earlier.
- **Dependency Mapping:** Builds visual maps of project modules and Salesforce components and enables the testing teams to focus attention on the important processes that are affected by changes and to allocate resources accordingly.
- **Smart Regression and Test Effect Analysis:** Effectively executes needed regression tests based on the impact analysis results and speeds up the validation process while decreasing irrelevant testing.
- **Real-Time Dashboards and Analytics:** Displays| dashboards that are live with respect to the status of impact assessments and testing for the project manager; can help inform data-driven go/no-go decisions.
- **Collaboration and auditability:** An audit trail of actions and analyses occurs together with collaboration in a transparent way for auditability and governance purposes.

- **End to end testing integration:** Automated pipelines brings together different types of testing layers (e.g., functional, integration, regression, performance, security or others) and predictive insight via tools to create frameworks for automated testing.
- **Project execution and team management:** Prepares the execution timetable for testing, writing the preparation of the testing plan, and managing defect reporting to create a systematic means of delivering against objectives through testing.
- **Advanced integration and API testing:** Using tools, advances testing of integrations and APIs.

Salesforce CI/CD testing consists of several important elements that can be done in several steps; first, configure a version control system (e.g. Git) to manage Salesforce code and metadata and facilitate collaboration with the use of branches; second, create a CI/CD platform (for example GitHub Actions, Jenkins, etc.) and utilize Salesforce APIs or CLI to build automation to interact with Salesforce; third, pull code by using Salesforce DX CLI commands to automate the organization and modularization of the repo; fourth, create different types of test suites (for example unit tests for components, integration tests for external interaction, and end to end tests for user flows, and more regression, security, and performance tests).

The scripts are established for build and test pipelines that will perform code analysis, execute automated tests, and create metadata to ensure that failure at any of these three steps will halt the build. Deployment would be automated to transfer the changes that were successful to Salesforce through available tooling options such as Salesforce CLI for incremental updates. Testing environments will utilize Salesforce sandboxes, and test data will be automated to ensure consistency of test conditions. There will be ongoing monitoring and feedback capturing mechanisms in place to provide live visibility and reduce the time to resolve issues. There will be a focus on iterating and optimizing based on performance data while maintaining governance and compliance through audit logs. All of this will lead to faster, more durable, and higher-quality Salesforce releases with a lower manual touch, alongside improved collaboration between development and QA [10].

It provides architectural support for the entire life cycle of the test life cycle and operates across environments and types of tests, offering automated and manual testing encompassing functional, integration, regression, performance, and security testing. It delineates scalability, traceability, and collaboration, supporting Agile and Waterfall methods in Figure 1:
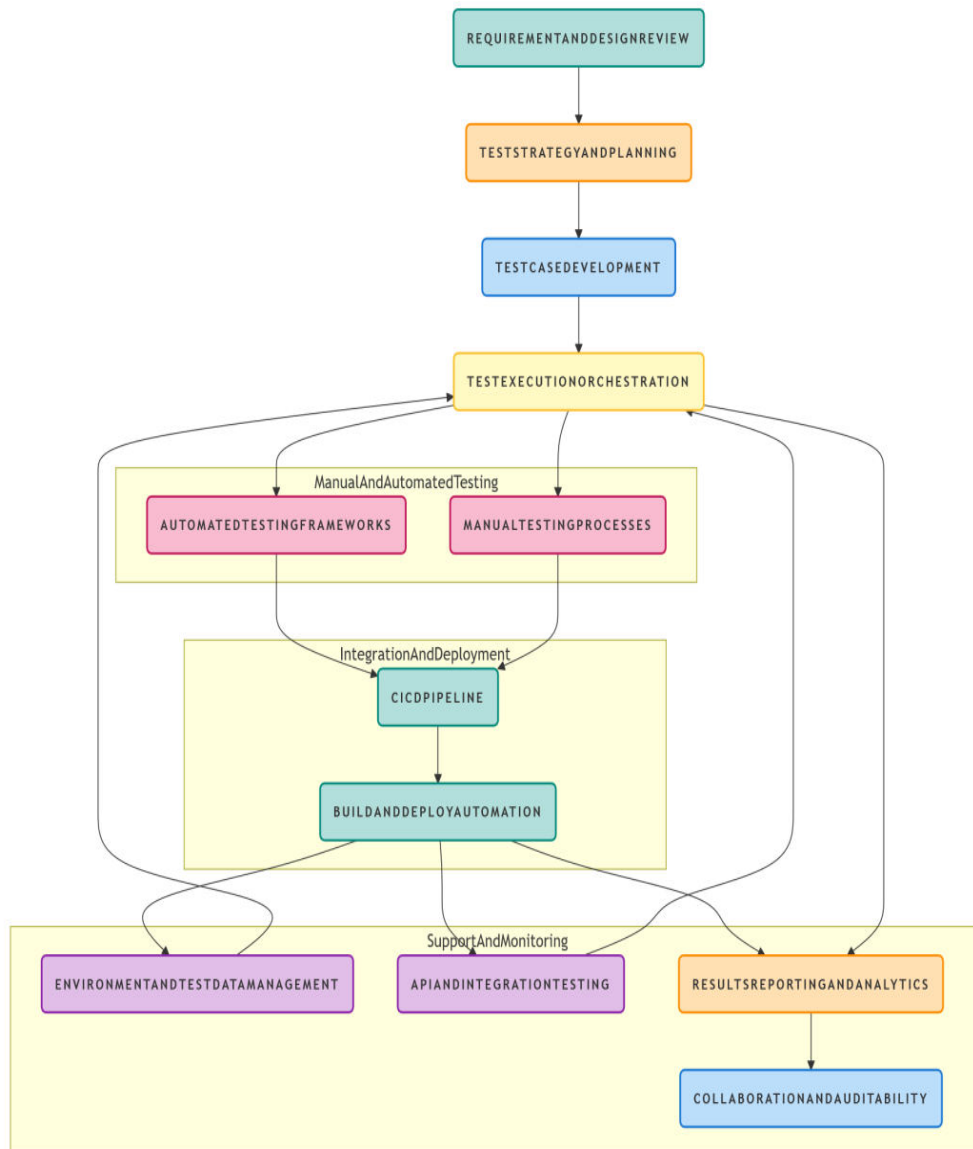
Figure 1: Salesforce Test Lifecycle Management Architecture

1. **Planning Layer and Strategy:**
   Several tools were used include Zephyr, Confluence, and Jira, which allow for an integrated approach to risk analysis, time and resource estimation, test strategy development, and requirement analysis.
2. **Orchestrating Test Executions:**
   Several tools were used including Provar/Testim, Jenkins/GitLab CI, Robot Framework, and Selenium and the functions include orchestrating automated and manual tests, regression, and user acceptability testing.
3. **CI/CD Platform:**
   The tools used in this layer were Azure DevOps, GitHub actions, Jenkins, and Salesforce DX CLI. These provided for deployments, static code analysis, automatic extraction of code, test suites upon pull request testing, and other integrations, with version control-Git-for execution across parallel sprints.
4. **Test Data and Environment Management:**
   This layer included the use of tools like data loaders, synthetic data generators, Salesforce sandboxes, and others for automated seeding and data masking for privacy, renewing and synchronizing environments, managing test data sets, and others.

5. **API and Integration Testing Framework:**
   Tools from the API layer were SoapUI and Postman, which are used to support system-to-system data flows to and from Salesforce, interface testing in Salesforce, and checking the accuracy of response from APIs.

6. **Analytics and Reporting Dashboard:**
   This layer was about the reporting and analytics dashboard and relied on many, many tools.

The architecture of the Salesforce project places a high value on holistic quality assurance that includes requirement reviews and risk assessment during the development of testing methodologies and plans. This involves continuous integration of code and information through automated pipelines for initiating test suites when the trigger conditions are met. Test orchestration technologies manage automated testing modes and manual testing modes for regression testing and user acceptance testing. Continuity of integration involves testing UI and backend simultaneously with API validations, as well as dynamic provisioning of test environments and data generally to keep testing conditions the same. Results display as metrics on reporting dashboards, and documentation and communications channels facilitate knowledge transfer and compliance. This whole process is ultimately intended to ensure proper testing coverage and regulatory compliance.

Some significant criteria for assessing the performance of Salesforce Continuous Integration / Continuous Delivery (CI/CD) testing are discussed in the following section. Test coverage is likely the most crucial metric for exposing the percentage of automation testing that has been performed on Salesforce settings and Apex code. The higher the coverage level, the less possibility there will be defects in production. The defect detection rate is a measure of the capacity of both automated and human testing to deliver testing at the point of the best opportunity for a product in production to achieve quality. The automation rate relates to the extent or percent of coverage of automated tests to offer fast, consistent, and scalable testing. Test execution time is the length of time to run a suite of automated tests, which provide quicker feedback, and offers room for iterative releases. Deployment frequency for changes made into production should occur more regularly as in standard release models; however, the more frequent releases, the more effective the CI/CD pipeline. Change lead time for moving the code from code commit to production deployment would be the final metric; and the shorter the time frame, the more agile a component of the capability. Change failure rate focuses on the rate of deployments released that resulted in issues; fewer failed deployments demonstrates quality and stability. Lastly, MTTR denotes maintenance time for resolving issues identified or discovered post-deployment.
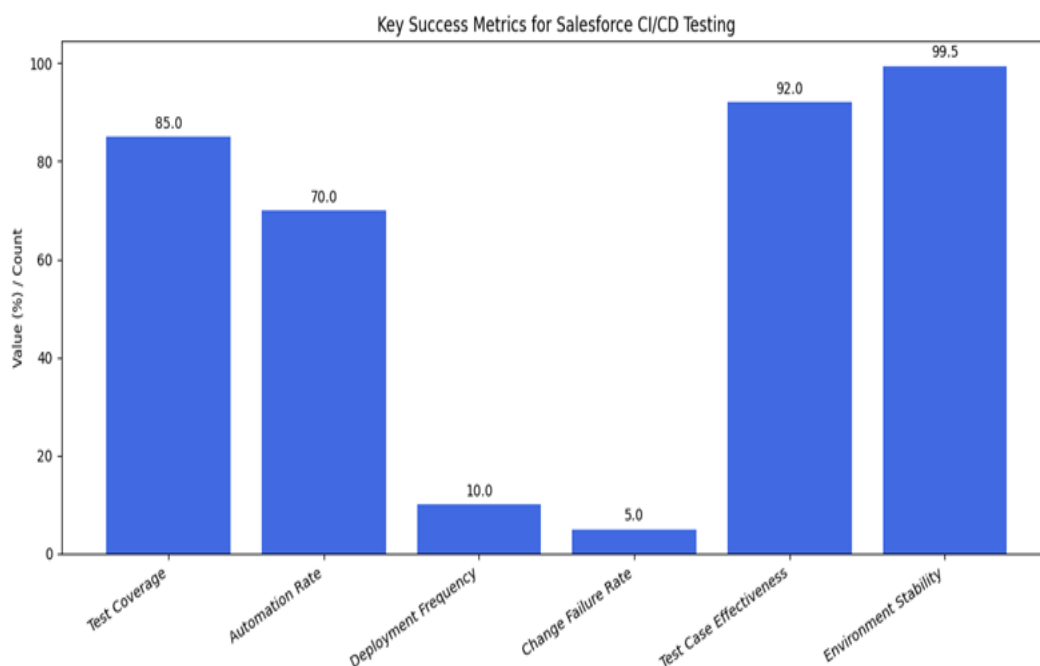

Figure 2: Key Success Metrics for Salesforce CI/CD Testing

IV. CONCLUSION

AI solutions for change detection in Salesforce metadata has greater advantages than human approaches, despite human tendency, speed, and accuracy improvements, as well as reducing false positives and alert fatigue from human error. AI solutions improve governance, increase operational speeds, and drive improvements in the release cycle. In addition, a human in a loop expert oversight is an all-important feature to regulated industries. In the future, artificial intelligence will make future strides in automating the de-risking of problem-solving, continuous monitoring, and oscillating in and out of integrations. These advancements will exponentially increase self-learning models with protection against more risk and lower cost. As capabilities advance and increase in scope, organizations will have easier relationship management of Salesforce metadata lifecycles, deploy intelligent connected QA ecosystems, and integrate approaches to continuous improvement to advance and respond to organizational and technology change. The advancements in automation and impact analysis will ease continuous integration and delivery with technologies such as Salesforce CLI and Jenkins. Future AI will adapt to continuously to the needs of the organization and further enhance the user experience through natural language processing. Overall, more organizations will find value and efficiency through flexible ecosystems that support fast delivery of value to the organization and innovation with Salesforce metadata management and testing.

REFERENCES

1. "Understanding Salesforce Metadata: A guide for non-techies", 20th October 2023, Rick Roesler, https://elements.cloud/blog/understanding-salesforce-metadata-a-guide-for-non-techies/.
2. "What is Salesforce Metadata? Definition, Types & Benefits", July 30, 2024, https://www.ibirdsservices.com/what-is-salesforce-metadata/.
3. "Salesforce Test Automation (Tools included)", May 14, https://www.browserstack.com/ guide/salesforce-test-automation-tools.
4. "Exploring AI Quality Assurance in the Agentic World: My Initial Thoughts", Nisha Jacob, https://www.techforceservices.com/blog/ai-quality-assurance-agentforce-testing/.
5. "3 Steps for a Successful Salesforce Data Migration", Lucy Mazalon, February 13, 2023, https://www.salesforceben.com/salesforce-data-migration/.
6. "What Is Salesforce Metadata? A Complete Overview", Shyam Oza, November 24, 2022, https://www.salesforceben.com/what-is-salesforce-metadata/.
7. "AI-Driven Automation of Salesforce Metadata Creation: A Technical Overview", Vani Panguluri, https://sarcouncil.com/download-article/SJECS-215-_2025-662-669.pdf.
8. "How to Build a CI/CD Pipeline for Salesforce", David Runciman, November 01, 2021, https://www.salesforceben.com/how-to-build-a-ci-cd-pipeline-for-salesforce/.
9. "Salesforce CI/CD best practices", David Runciman, July 7, 2023, https://gearset.com/blog/salesforce-cicd-best-practices/.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462   6381 907 438   ijircce@gmail.com

Scan to save the contact details