



autorabit

Continuous Delivery for Force.com

Achieve higher release velocity (shorten release cycles)
& reduced Time to Market by 40%

TO LEARN MORE ABOUT AUTORABIT,
VISIT [AUTORABIT.COM](https://www.autorabit.com) OR CALL +1 925 500 1004

REQUEST A TRIAL

Continuous Delivery for Force.com

Release Management at the Enterprise Level

At enterprise level, releases are integrated and sandboxes are synchronized across geographical locations, so the multiple product integrations and products can be released as a whole. Large organizations have complex development processes and multiple release cycles in a day. It aims at increasing productivity and reducing change by executing the release of related features together.

Why software releases are harrowing

Releasing software is predominantly a manual process, which leads to less frequent releases encompassed by change management systems. Development projects can span multiple sandboxes across various geographical locations, and the code needs continuous tracking and multiple changes, in which case the complexity of integrating code lines is increased. This makes the software release process sluggish and complex.

Understanding Release Management and its major aspects

Release Management is the art of building, testing, packaging, and deploying software for consumption. It is effective for organizations that have multiple products which work together to get a solution. Multiple orgs will either need their own release management process or the existing release management process will become much more complex.

The goal of release management is to preserve the integrity and availability of production systems during the deployment of new software or hardware and the process of managing software release from the development stage to the final release of the product.

Agile method of software development has led to faster release of the products by organizations. To deal with the continuous cycles of development, testing and release the release management method has to be adopted to avoid any last minute glitches during the final release of the product. It has the task of designing testing and installing pre defined changes in a live environment.

Release management in large projects requires tracking changes between the various environments to ensure that there are no clashes when the deployment is done.

Through 2016, a lack of effective release management will contribute up to 80% of production incidents in large organizations with complex IT services.

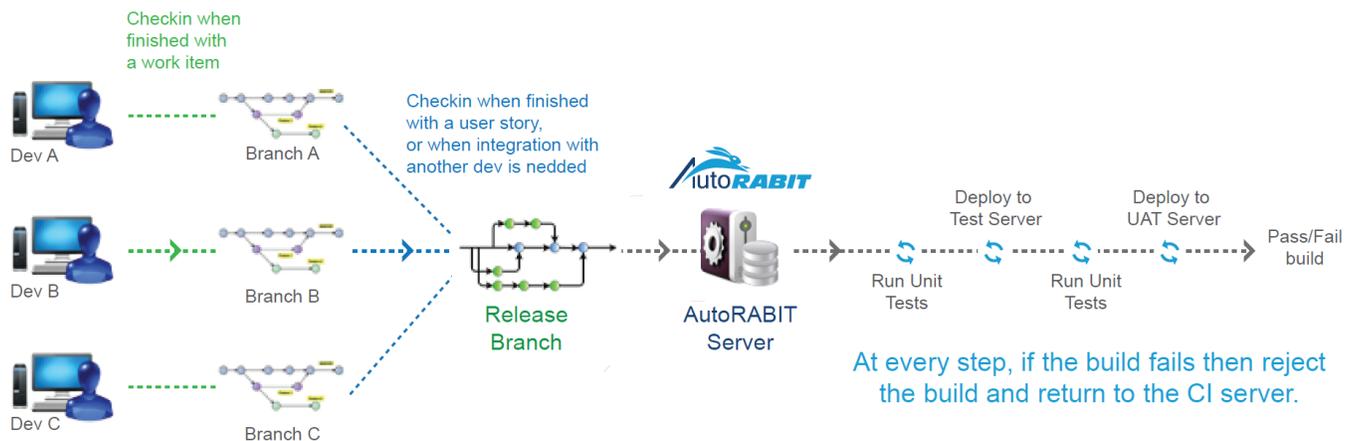
- Gartner

Face of Release Management

Continuous Integration in Salesforce

Continuous Integration is integrating the metadata into a single shared repository or a source control like GIT, SVN, TFS. Integration can be done several times a day depending on the need of the project. This is a very crucial practice in the agile method of development. Following the method of continuous integration has many benefits like; earlier bug detection which helps the developers in spending more time to add features to the product. It increases the visibility, which helps in faster development and in turn helps in delivering the product on time.

Let's look at how Continuous Integration is enabled for salesforce using AutoRABIT.



Continuous Deployment in Salesforce

Continuous Deployment is pushing the metadata into production that is deployment ready.

It is the process of releasing the metadata into production after the testing is completed. It is an extension of continuous integration. It minimizes the lead time between the actual code development and the product being used by the end user. After each successful integration, that is, after the code meets certain guidelines or release criteria, the actual application are updated with the new code. This reduces the time in getting the feedback from the customer after each new feature of the product is released. This helps in earlier return on investment for the customer which considerably reduces the capital investment on the product.

Let's look at how Continuous Deployment is enabled for salesforce using AutoRABIT

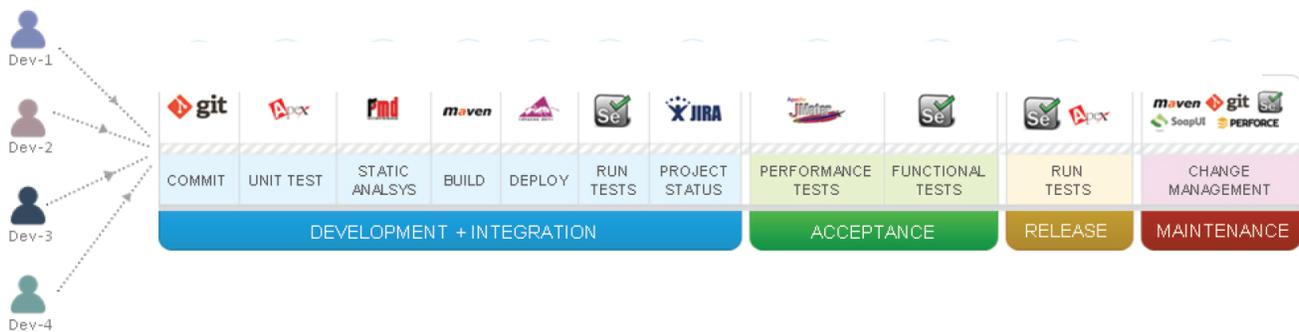


Continuous Delivery in Salesforce

A medley of Continuous Deployment and Continuous Integration.

Continuous Delivery assists in higher release velocity and increased Time to Market. The teams have to ensure that every change made to the software is releasable, by which the team can ensure that the product is released on time. This makes the releases more frequent and customer feedback can be taken, the changes can be made accordingly.

Let's look at how Continuous Delivery is enabled for Salesforce using **AutoRABIT**.

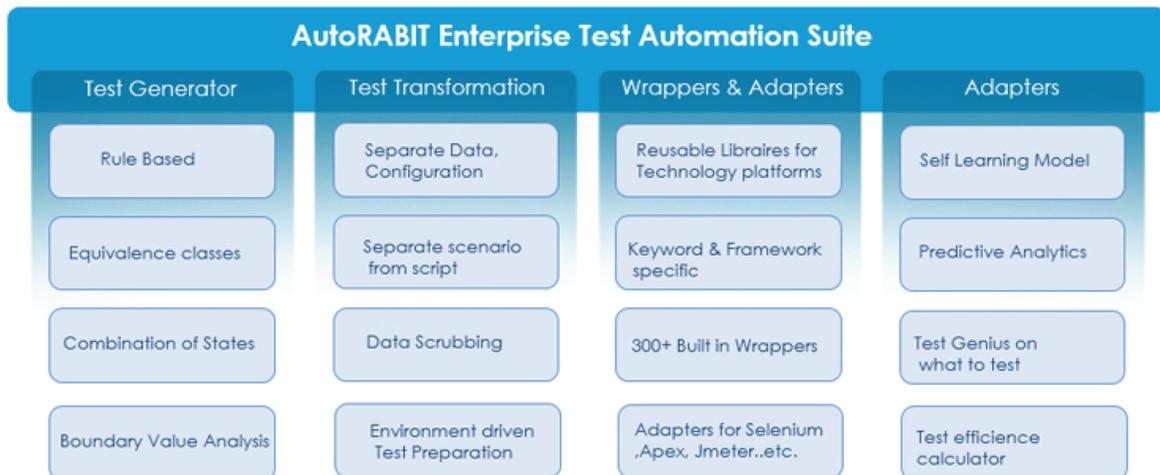


Test Automation

Software tools executing tests on a chunk of code is test automation.

Repeated testing might be required during the testing process. Here automated testing comes to the rescue. Once developed, automated tests can be run quickly and repeatedly. Thus reducing the time required for testing, which in turn increases the speed of the development of the product, decreases cost, and also gives more time to market the software developed. Sometimes even small or minor defects can cause the entire software to breakdown. So this is a cost-effective method for software products which have a long maintenance life. There are many software metrics that are used to determine the condition of the software or the adequacy of the testing.

Let's look at how Test Automation is enabled for Salesforce using AutoRABIT.



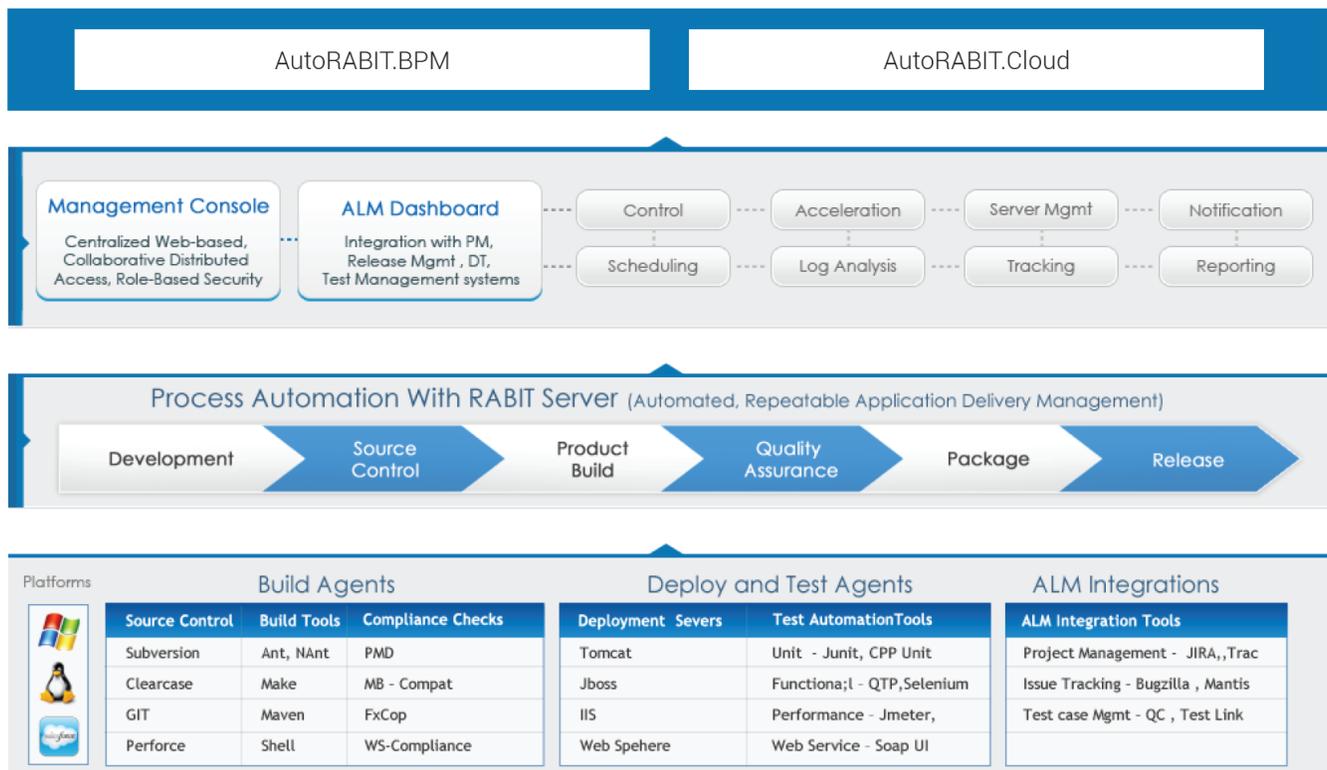
AutoRABIT a Release Management suite for Salesforce

Continuous Integration in Salesforce

Update the existing software development workflow and ensure selective feature deployment, synchronization of geographically spread sandboxes, synchronization of code deployment, release tagging and committing to SVN.

AutoRABIT is compatible with industry's leading version control and defect tracking systems. And, all these features can be used from a single user-friendly ALM dashboard and all these are available in public and private Clouds.

Release management is not completely automated in Force.com. Manual release management is a herculean task and not always reliable as it is prone to human error.



Features in AutoRABIT

Version Control

Every change made to the code is tracked, along with details like, who made the change, why they made it, and references to problems fixed, or enhancements introduced, by the change.

Feature based custom deployment

By using AutoRABIT the user can update the existing software development workflow and it ensures selective feature deployment, synchronization of geographically spread sandboxes, synchronization of code deployment, release tagging and committing to SVN.

Automated Deployment Schedules

The metadata deployment can be scheduled as well as automated. There is an option of “one-click” promotion of sandbox’s metadata to different release environments.

Test Generation and Test Automation

In AutoRABIT the tests are categorized in context of application sanity and also enable easy feedback cycles. It lets the team prioritize their test plans so they can execute their tests effectively ranging from sanity checks to exhaustive testing. The user can also view the detailed error reports which make it easier for the user to track the errors clearly.

Multi Browser Testing

Multi browser testing is important to avoid alienating customers using different browsers. (Internet Explorer (6, 7, 8): 30.4%, Firefox: 46.4%, Chrome: 16.7%, Safari: 3.4%, Opera 2.3%) This is a process to check whether the application is functioning across multiple browsers. In AutoRABIT there is an option of multi browser testing, which helps in determining if the application is compatible with multiple browsers.

Apex Code Coverage

At least 75% of the Apex code must be covered by unit tests, and all of those tests must be completed successfully. A panel displays the code coverage percentage for every class in the organization.

Integrated Salesforce Dataloader

AutoRABIT has a powerful Data Loader component for Salesforce which clearly stands out ahead of traditional standalone manual data loaders today. It is a web based tool. Login to the Org, Choose the objects [Standard and custom objects], choose the operation. Check for metadata integrity between the source and the destination and get detailed reporting in case of any failure at the time of data loader operation

Sandbox Management

Releases can be easily synced across geographically spread sandboxes. The version controller analyzes detailed changes after every cycle and handles the SCM administration. The sandbox comparison and sandbox management restores the configuration of the testing environment after every refresh.

AutoRABIT synchronizes and adapts the workflow with your present processes and team evolution. AutoRABIT helps make revenue growth possible and the release more seamless for the product and ensures successful job completion.

ALM Dashboard

Assistance is provided by the advanced ALM dashboard which integrates key information like requirements, progress, and defects. The test automation is fine-tuned by change analysis, code testing, and defect logging, for out-of-the-box SFDC applications. Availability of cross browser testing option increases the confidence of the test results.

Achievements using AutoRABIT

DevOps collaboration

Organizations using DevOps can deploy a code 30% more frequently. DevOps is basically a combination of Development, Quality assurance and Technology operations. It promotes a set of processes and methods for communication and collaboration between different departments of an organization. It aids in software application release management by standardizing the development environments. Many organizations have the pressure to do quick releases in response to the requests coming from the users or the clients of that particular product. Here the development teams have to support the operational requirements like deployment, load, performance testing and so on from the beginning of the cycle and the operations team has to provide support and feedback before, during, and after the deployment of the code. So DevOps basically bridges the communication gap between the development team, QA team and the operations team.

Time to market

From Idea to Delivery
Time to market is the period of time between the inception of an idea to the availability of the product in the market. If more time is taken for the development of the product, it will impact the organization in many ways, like less customer satisfaction, loss in market share and increased complexity of the delivery process. Improving the time to market will have a very positive effect on the profit margin and market share of the product.

