VIEW POINT

PRESCRIPTION FOR PHARMA IT LEADERS TO DELIVER FASTER AND SCALE EFFICIENTLY

Infosys®
Navigate your next

Business Need for DevOps and CI/CD in Life Sciences Industry

Life science companies typically generate large volumes of data associated with research, product development, clinical trials, manufacturing, supply chain and post market surveillance. There are regulatory, legal, and country specific requirements to retain these data sets over a long period.

The IT infrastructure and applications required to support the business needs of such large volume regulated enterprises are always evolving, more so becoming large and complex in tandem with the growth of the organizations. Many a times, the IT divisions of life science companies are constrained in their ability to scale up and meet business needs quickly due to traditional methods of system development and implementation, and documentation-intensive Computerized System Validation requirements.

The problem statement then for IT leaders is three-fold:

1. How to improve the speed of delivery?
2. How to scale in a cost-effective manner?
3. How to bring about change in a highly regulated environment?

Improving the speed of delivery and scaling in a cost-effective manner requires a re-thinking of the People structure, Processes being followed and the Technology that is being used.

In this context, it would be worthwhile defining key concepts of DevOps, Continuous Integration and Continuous Deployment (CI/CD) before proceeding to discuss how they can solve the problem statements for IT leaders.

DevOps and CI/CD Tools

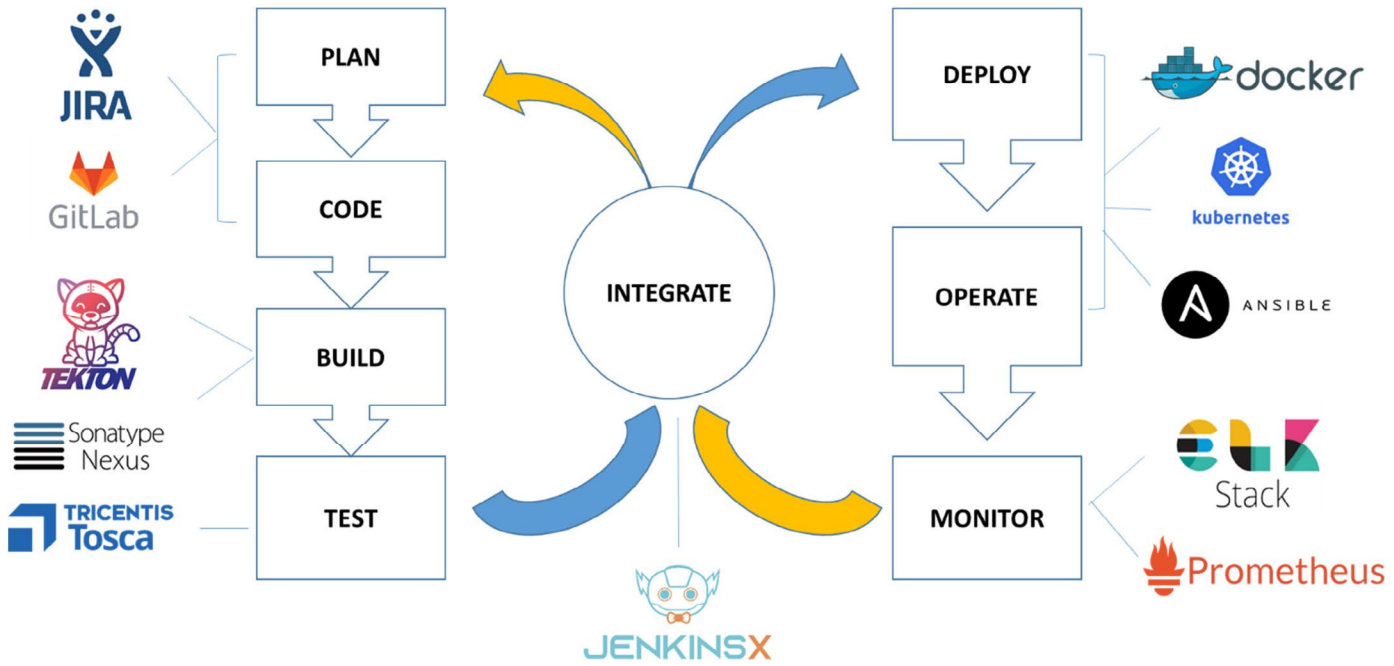
DevOps is a set of operating principles and collection of practices that enable application development teams to deliver code changes more frequently and reliably. Under a DevOps model, an integrated team of engineers works across the application lifecycle (build / test / release / monitor) using automation tools to deliver changes faster to the end users.

Continuous Integration and Continuous Deployment are individual best practices within DevOps. In the context of an application enhancement, changes in scope of development are split between multiple engineers and each individual development work is identified via a development branch. Multiple development branches merge to form a master branch. As part of Continuous Integration, engineers regularly merge their code changes into the master branch where a 'development build' is generated and tested. Based on the automated test results, code changes are made in the development branch and merged again in the master branch.

The final build is demonstrated to business users via demo session as a best practice before it is pushed to the formal regulated environments. Continuous Deployment is an extension of Continuous Integration, but it doesn't stop at the test phase. Continuous Deployment is all about going from code changes to build, testing and subsequent deployment to production in an automated manner. The idea behind the two best practices is that small incremental changes are pushed to production via frequent releases using automation in a reliable manner.

Several tools in the market provide CI/CD capabilities. Organizations can either choose a suite of proprietary tools from established vendors (e.g. Microsoft Azure, AWS DevOps etc.) or identify a combination of open-source tools based on the need, ease of use & setup, integration capabilities, hosting options (on premise / cloud), in-house capabilities, available budget and OS supports. Some of the key advantages with open source tools is the possibility of having an increased level of control over customized on-premise tools (at a low investment cost) that also integrate well with the existing complex regulated corporate landscape. In both cases of proprietary and open-source, a collection of tools would be required to meet the objectives of planning, code management, building, testing, integration, deployment and monitoring.

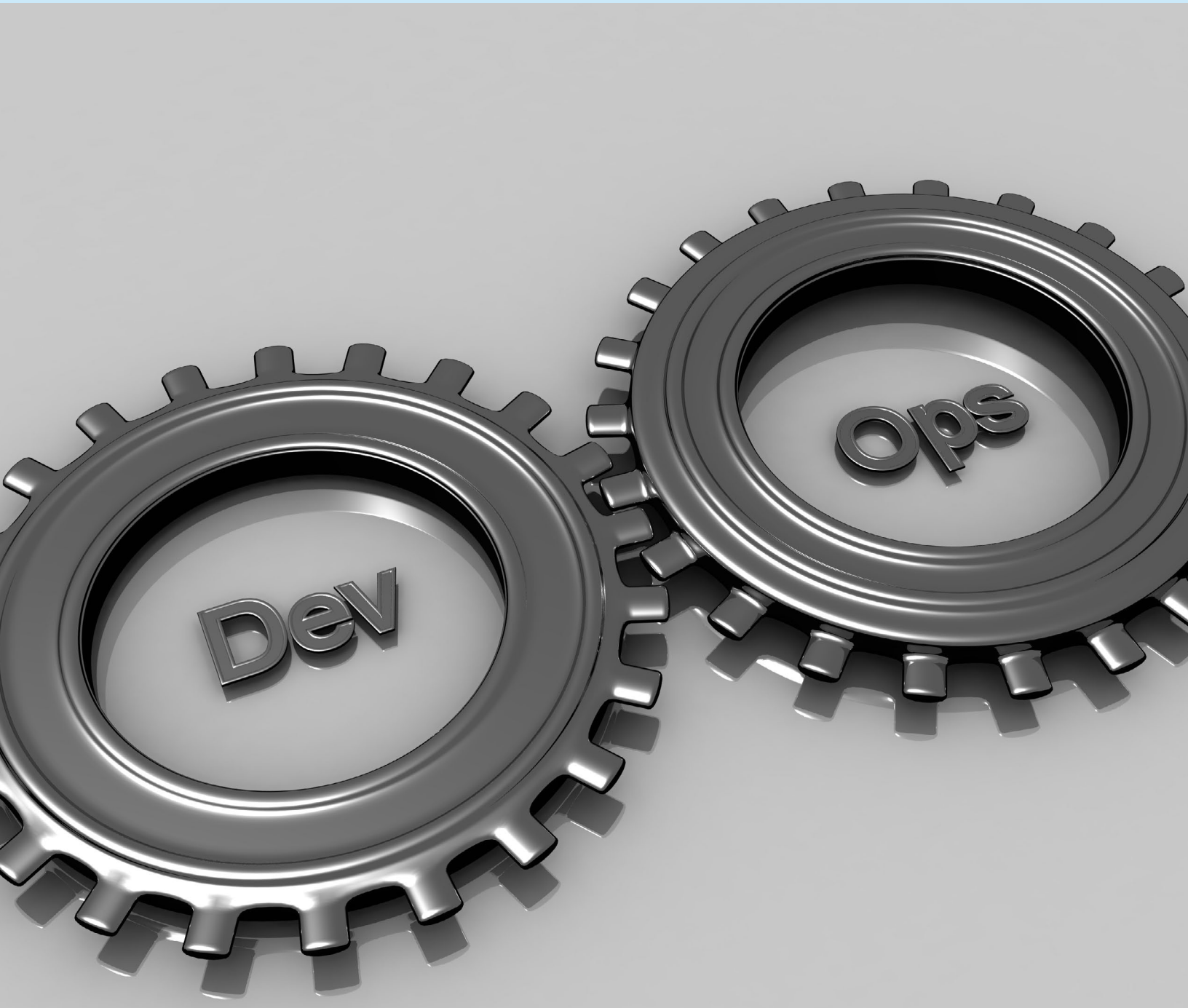
An example of a core set of open-source CI/CD tools that can be used by an organization are shown below.



With the toolkit shown above, application code changes are managed in GitLab and the committed code changes are pushed through a deployment pipeline that is managed by JenkinsX. Test automation is managed using Tosca. Dockers, that are equivalent of virtual machines, provide the layer for application and middleware components. Multiple dockers could be setup to support the same application. Kubernetes manages the dockers to ensure continued availability of the application. The entire process from pushing the code change to application deployment is completely automated.

Development of application changes following traditional methods and subsequent deployment and testing with formal documentation could take anywhere between a few weeks to a couple of months. With CI/CD tools, as most of the activities are automated, the timeline for deployment and testing is reduced to a few hours. Automation also leads to rationalization of IT resources. With changes deployed faster, IT divisions can scale quicker, in a cost-effective manner and meet growing business requirements faster.

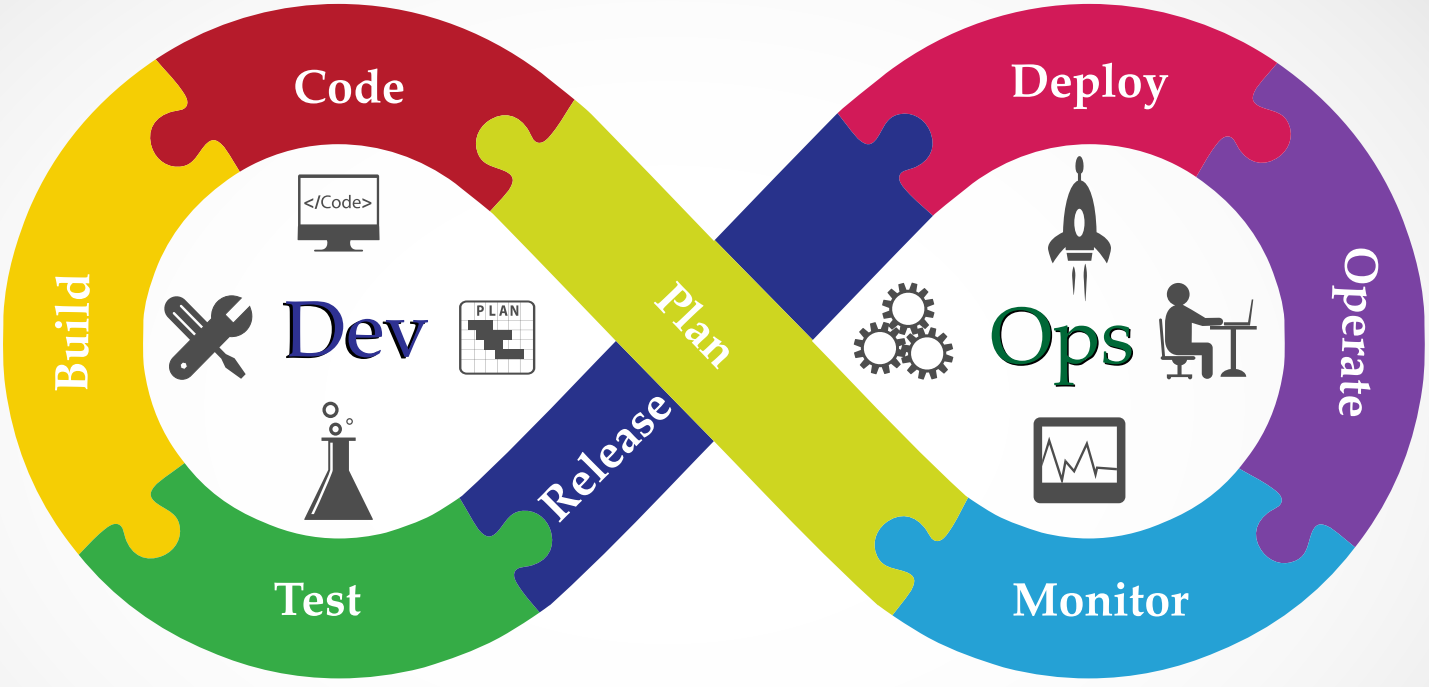
Stepping back from the technology aspect, there are still two more challenges to be looked at: People structure and Processes followed in life sciences industry.



Organization Setup

In a traditional setup, IT organizations have a matrix structure where resources are shared across systems and projects based on requirement. In this setup, the organization's view is focused on completing itemized tasks (as projects) within a defined budget and a definitive timeline. Once the project is completed, resources are released and to a certain extent, knowledge of the system is lost. Subsequent enhancements require appropriate business justification for funding procurement. Any improvement in ways of working or automation would additionally have to be justified and may be an overhead for business that may not be willing to fund such activities.

In such scenarios, taking a long-term view on systems as "business assets" with continuous incremental operational enhancements would fit business needs better as opposed to going through infrequent operational enhancements as projects. In this view, resources can be aligned based on a system or a product. For every system, there would be Product Manager, Developer, Tester, Quality Manager, Release Manager, Operations Expert and a possible representative from Security team. The roles can be further mapped to application development lifecycle of Build / Test / Release / Monitor. This structure gives the management a better view of what is required from resource perspective to support the operations of systems. Here, business would also be more committed to funding over a long period when the changes are small, periodic and consistent. In this structure, knowledge of the system is retained within the team enabling the team to work closely with business and address their requirements. This kind of a team structure coupled with CI/CD tools would ensure business needs are addressed faster and in a consistent, reliable manner.



Regulatory Processes

In high-tech industry, CI/CD tools can be used for automated build deployment to the Development, Test and Production environments. However, in life sciences industry, Test and Production environments are subject to rigorous control processes. Any changes in scope of deployment in the regulated environments are subjected to a thorough impact assessment, review and approval by multiple teams before it is approved for deployment. Even then, there are strong checkpoints on the levels of testing conducted in the Test environment before the change can be deployed in the Production environment. Other processes such as Incident and Problem Management also have similar rigor. Proper testing, documentation, review, and approval of test documents is required before incidents or problems can be closed. In such a scenario, the full benefits of CI/CD automation cannot be fully achieved by life science companies if their processes are written with waterfall or agile methodology as the backdrop. Therefore, to maximize the benefits of CI/CD, existing processes have to be adapted. For instance, organizations can look at simplifying the change management approach by mandating fewer manual approvals. To facilitate this, the CI/CD tools have to be validated thoroughly to ensure that the logs generated are truly reliable, tamper-proof and the levels of testing are worthy of withstanding audits.

Bringing about Change in a Regulated Environment

Though there are clear advantages with DevOps and CI/CD adoption, moving away from traditional project delivery methodologies or adopting new technology may not always be easy in a regulated environment like that of the life sciences industry.

In many organizations, the need for such changes are triggered in a few projects and are usually tackled on a case-to-case basis at the project level before they are considered for enterprise wide adoption. This approach is time intensive and is not the best suited in our opinion to scale efficiently and faster. Organizations should be structured to be agile and be open to challenge the status quo. This is fundamentally a question of the organization's culture and mindset.

As with all enterprise wide changes, for widespread adoption of DevOps and CI/CD, there should be a top down concerted effort to champion the benefits of the tools and methodology via series of discussions so that awareness is built and the cultural shift from manual repetitive operational processes to a more nimble automated operational process happens. This requires a strong leadership team that is able to define and communicate the vision, goals and objectives effectively, and rally the troops towards achieving the objectives by providing them with the necessary support required in making the changes.

Implementation of CI/CD tools and DevOps methodology also requires strong technical resources who can bring together the experience of using these tools and methodology in other fast-paced industries and can integrate them to work effectively in a regulated environment. Management support in bringing on board the right technical resources (from developers to senior technical leaders) is also crucial in change adoption as teams with the right experience produce high quality output with lesser errors and re-work.

In conclusion, we propose that an IT organization structured based on systems or products, adopting DevOps methodology and CI/CD tools, will improve the 'go-to-market' time thereby enabling IT leaders to deliver faster and scale efficiently.

About the Author

Dinesh Amirtharaj Peter, *Senior Consultant, Infosys Consulting*

For more information, contact askus@infosys.com

Infosys®
Navigate your next