DevOps: Helping Maximize the Benefits of Agile Delivery

Executive Summary
Enterprises today need to rapidly evolve to meet customer requirements, combat competition, adopt disruptive technologies, and manage budget cuts, to sustain and grow in this post crisis era. With business agility emerging as a key requisite to staying ahead of the competition, enterprises have adopted Agile methodologies to enhance the pace of software development and align with changing requirements. However, these initiatives sometimes fail to deliver the desired benefits, mainly because of the absence of collaboration among different stakeholders. This necessitates changes to the standard approach to achieve the objectives of faster time to market and higher return on investment. This paper examines how Agile delivery methodology can be extended to resolve key issues.

Introduction
A dynamic market characterized by rapidly changing customer needs is putting immense pressure on enterprises to launch solutions in a timely manner to retain market share. To operate successfully in such an environment, enterprises need to be agile enough to take advantage of new opportunities. Besides using technology to automate their systems and processes, enterprises have typically embraced Agile software delivery methodologies to reduce the time to market their solutions. However, the benefits derived from these efforts are often limited because of a lack of collaboration among the different functions within an organization. Enterprises should therefore review and refine their approach toward transformational Agile initiatives to completely realize the related benefits and maximize the return on investment.

Agile Adoption in Enterprises
Today, information technology (IT) plays a critical role in the efficient functioning of an enterprise. This dependence often translates into enormous demands on the IT teams to accelerate the delivery of IT applications. As a result, enterprises constantly strive to hasten the pace of software development by adopting emerging technologies and methodologies such as Agile delivery practices. Agile methodologies advocate incremental development based on continuous feedback from the product owner, which helps identify issues earlier in the delivery lifecycle thus enabling adjustments to the product design based on the changing requirements. This dramatically reduces development cycle times, cuts costs, enhances quality, and significantly shortens the time to market.

Widespread Agile adoption in enterprises, however, has not resulted in the delivery of fully operational software within the specified timelines. Siloed functioning of the development and operations teams has led to designers, developers, testers, support staff, and the infrastructure provisioning teams working with little or no collaboration. More often than not, the development teams churn out code at regular intervals and these ‘potentially shippable products’ accumulate before the operations team is able to release them for production. When the operations team eventually releases the software application, it often fails to function as expected in the production environment. This is usually due to the failure in meeting the non-functional requirements when the application is in production. Enterprises need to design and implement internal processes and operating procedures in a manner appropriate to achieve continuous deployment of working software.
Key Challenges Faced by Enterprises

Based on our experience with enterprises using Agile methodologies, we have identified some key issues faced by organizations that function in silos. These issues cause delays in solution delivery, software failures in production, and end user dissatisfaction.

Time lag between development and deployment

Development teams working in the Agile Scrum model deliver functionality incrementally, that is, at the end of each sprint. However, this functionality is not deployed into production immediately because of multiple constraints including the lack of readiness of the operations team. The development team limits its role to delivering the functionality after obtaining sign-off from the product owner, and generally moves to the next sprint without focusing on how the completed deliverable subsequently progresses towards deployment.

The delay in deploying the functionality to production can be attributed to the following reasons:

- Inability of the infrastructure team to provide the target deployment environment (release/test and production environments) according to the specifications of the operations team
- Poor configuration management
- Absence of operating manuals and the consequent lack of information to end users about usage, maintenance, and troubleshooting
- Lack of training of production support personnel resulting in poor servicing of L1 and L2 support

Minimal collaboration among teams

The lack of collaboration among different teams manifests in the following ways:

- **Lack of understanding of performance requirements**: Development teams write the software code without a clear understanding of the non-functional requirements that the end product should meet. As a result, the teams fail to architect the underlying technical components of such requirements during design, and build them to align with the operating environment. This disconnect between the development and operations teams results in poor quality solutions, incapable of satisfying the specified parameters related to response time, uptime, the time required to restore functionality during a disaster, and so on.

- **Disparate environments**: The development teams, functioning in silos, write the code and test it in the development or quality assurance environment that typically uses different tools and methodologies and does not mirror the release testing or production environment in entirety. Subsequently, when the operations team tests the product in the release or production environment, the application often fails, necessitating several changes to the software. These unanticipated changes can interfere with the smooth functioning of the production systems, further delaying deployment and solution delivery.

- **Multisite deployment issues**: Global enterprises require applications that can be deployed at multiple locations and with multiple language interfaces. This requires the development team to consider the inherent geography specific constraints and cultural differences while configuring the application. Such limitations could include factors such as low telecom maturity, slow network speeds, and linguistic and cultural diversities.

- **Lack of alignment between business and IT**: Developers are quick to embrace the latest technologies and tools. But they often fail to adequately focus on the requirements of business continuity, scalability, reliability, and repeatability. In conjunction with insufficient knowledge about the impact of an application on business, infrastructure, and end user experience, this divide results in poor quality applications that neither meet user expectations nor deliver business value.

Addressing the aforementioned issues, and breaking down the silos in traditional software development environments, is vital to increasing efficiency. Siloed functioning of the development and operations teams impedes the deployment of IT solutions despite the adoption of Agile methodologies. This involves a paradigm shift in the approach to software development and the adoption of a collaborative style of functioning that encourages communication among teams.

An Approach to Resolve Key Issues

The disconnect between the development and operations teams, and the associated problems, gave birth to the concept of DevOps, which focuses on eliminating the barriers between the two functions by bringing the two teams together. In the DevOps model, the development and operations teams work in sync through the software development lifecycle. The DevOps culture involves collaboration among teams right from ideation to design and build, and test to production. By enabling the DevOps way of working, enterprises eliminate silos, remove waste (non-value adding activities such as redundant process steps, unnecessary documentation and so on), lower development cycle times, and quickly deploy solutions into production to significantly reduce time to market.
DevOps necessitates a fundamental shift in the approach to software development and resolving IT problems. It requires developers to collaborate with operations to gain a clearer understanding of operations related activities including performance testing. Further, the operations team provides continuous feedback and draws developers’ attention to how the application impacts infrastructure, and functions in real world environments. The operations team works in tandem with the development team to sort out IT issues, thereby facilitating the development of high quality software. By creating a culture of two-way communication between the development and operations teams, DevOps bridges the gap between these two key functions.

We recommend that enterprises adopt a DevOps way of working in their Agile environments to achieve the continuous delivery of working software.

A Practical Approach to Establish a DevOps Environment

The four fundamental tenets to a successful establishment of a DevOps environment are — culture, automation, collaboration, and metrics.

**Culture:** Incorporate a DevOps mindset within the enterprise, with a focus on fostering collaboration to achieve shared goals. In the absence of a collaborative organizational culture encompassing people and process, automation initiatives will fail to yield the desired results. The DevOps way of working establishes a culture that eliminates hierarchical interactions, promotes open communication, and encourages feedback. In addition, such an environment motivates people to ask questions and find answers, encourages knowledge sharing, and removes the fear of failure.

**Automation:** Automate routine, predictable tasks such as build and integration, regression testing, configuration management, infrastructure provisioning and configuration, and application deployment to rapidly deliver high quality software. Deploy automation tools that help bring the development and operations teams together, thus ensuring continuous deployment.

**Metrics:** Establish a mechanism to define, analyze, and report metrics to identify leading and lagging indicators that can trigger improvement initiatives.

Typical metrics include, delivery cycle time, frequency of deployment, application performance, application stability, mean time to recover, service level agreements, defect density, operations cost, end user experience and customer satisfaction index.

**Collaboration:** Collaborate throughout the software development lifecycle, share knowledge, best practices, provide feedback on performance, troubleshooting, and so on.

Implementing DevOps practices in an Agile Environment

Successfully introducing the DevOps way of working into an Agile development environment depends on the following key factors:

- **Involve the operations team through the development lifecycle right from ideation through build, test, and continuous deployment.** The operations team should provide inputs on requirements related to the deployment environment, security or firewall, database backup, and restoration needs to ensure these requirements are built into the product design.

- **Automate build, integration, release, and deployment activities.**

- **Facilitate collaboration between the product owner and the operations team to ensure all stakeholders have a clear understanding of the non-functional requirements.** Product owners should collaborate with the operations and development teams to gather inputs related to deployment and support.

- **Create exclusive product backlog stories that specify performance requirements, and capture the non-functional requirements including operating instructions and manuals, guidelines for rollback and roll forward, and so on.** Coordinate with the operations team for infrastructure provisioning and plan for the necessary investment.

- **Ensure the Scrum team has appropriate representation from the operations team for everyone to get familiarized with the others’ nature of work and associated challenges.** Large organizations can consider job rotation to achieve this. Ensure participation of the operations team in sprint planning, sprint reviews, and sprint retrospectives for them to provide inputs on operational aspects.
Expand the scope of Definition of Done (DoD) to include validation of code in near-production environment, completion of support instructions, and dry run of these instructions.

Align the enterprise release schedule with sprint plans based on the sprint completion pattern.

Track key metrics including percentage of timely releases, increase in number of releases, proportion of defects that can be attributed to operations related issues, and so on.

By instilling the DevOps way of working into Agile environments, enterprises can increase revenue and profitability, enhance application quality, reduce cost of operations, shorten development cycles, reduce the time to market, and improve IT service management.

Conclusion

Enterprises have failed to fully realize the benefits of Agile delivery because of the gap between the development and operations functions. Implementing the DevOps approach to software development can help enterprises achieve continuous delivery of working software thereby enhancing end user satisfaction and maximizing the benefits of Agile delivery.

About the Author

Ravishankar N

Ravishankar N works with the Global Consulting Practice (GCP) unit at Tata Consultancy Services (TCS) as the competency lead for Agile and DevOps. With over 20 years of experience in the IT industry, including 13 years in IT process consulting, Ravi provides advisory services to global organizations in designing and implementing quality and process improvement initiatives involving Agile and traditional methodologies. He also develops consulting offerings around DevOps and has created an offering to enable an IT organization assume an agile way of working within a short period of 90 days. In addition, Ravi has published several papers in research journals and industry forums.
About TCS’ Banking and Financial Services Business Unit

With over four decades of experience working with the world’s leading banks and financial institutions, TCS offers a comprehensive portfolio of domain-focused processes, frameworks, and solutions that empower organizations to respond to market changes quickly, manage customer relationships profitably, and stay ahead of competition. Our offerings combine customizable solution accelerators with expertise gained from engaging with global banks, regulatory and development institutions, and diversified and specialty financial institutions. TCS helps leading organizations achieve key operational and strategic objectives across retail and corporate banking, capital markets, market infrastructure, cards, risk management, and treasury.

TCS has been ranked #2 in the 2014 FinTech Rankings Top 100 of global technology providers to the financial services industry, by both - FinTech Forward™ (a collaboration of American Banker and BAI) and IDC Financial Insights. TCS has also been recognized as a ‘Leader’ and a ‘Star Performer’ in Everest Group’s 2014 PEAK Matrix reports for Banking and Capital Markets Application Outsourcing (AO).

About TCS’ Global Consulting Practice

TCS’ Global Consulting Practice (GCP) is a key component in how TCS delivers additional value to clients. Using our collective industry insight, technology expertise, and consulting know-how, we partner with enterprises worldwide to deliver integrated end-to-end IT enabled business transformation services.

By tapping our worldwide pool of resources – onsite, offshore, and near-shore – our high caliber consultants leverage solution accelerators and practice capabilities, balanced with our knowledge of local market demands, to enable enterprises to effectively meet their business goals.

GCP spearheads TCS’ consulting capacity with consultants located in North America, UK, Europe, Asia Pacific, India, Ibero-America, and Australia.

Contact
For more information about TCS’ Global Consulting Practice visit:
http://www.tcs.com/offerings/consulting/Pages/default.aspx
Email: global.consulting@tcs.com

About Tata Consultancy Services Ltd (TCS)

Tata Consultancy Services is an IT services, consulting and business solutions organization that delivers real results to global business, ensuring a level of certainty no other firm can match. TCS offers a consulting-led, integrated portfolio of IT and IT-enabled infrastructure, engineering and assurance services. This is delivered through its unique Global Network Delivery Model™, recognized as the benchmark of excellence in software development. A part of the Tata Group, India’s largest industrial conglomerate, TCS has a global footprint and is listed on the National Stock Exchange and Bombay Stock Exchange in India.

For more information, visit us at www.tcs.com

IT Services
Business Solutions
Consulting