DEVOPS ENGINEERS CHALLANGES

- **Problem Resolution:**
- One of the essential advantages of DevOps is that it does not waste any time. Rapid Updates and Deployment are enabled by aligning the resources and people of the company. The DevOps programs fix problems before they become worse. DevOps create collaboration among security teams, operation teams, and development teams. DevOps also promote a culture of transparency in the organization.
- DevOps allows the problem to be solved faster because the ability to trace anything is very high. One has more confidence in the visibility and the delivery of the operations.

- **Time to Market:**
- DevOps is essential to make the process more straightforward. A business process is converted from a complex to a simple process. The time is taken to complete the process, thus shortens extensively. This allows the organization to be more responsive to the needs of the customers, receiving feedback faster on the features and more time to do marketing.
- **Reduction in Cycle Time:**
- DevOps provides more agility to the development of software. It helps in the delivery of code with insight. The process of DevOps should be well-crafted, and gates should also be there. The current version of the software application can also run side by side with the new version that you are going to deliver.

- A comparison of various metrics, such as performance metrics can also be made to know if the development is achieving the objective and goal of developing.
- Faster cycles of release and continuous improvement are promoted in the team of development by DevOps. It helps one to spend lesser time on the management of technology, processes, and tools and focuses more on other important matters such as providing better user experience.
- **Deliver value to Customers:**
- DevOps minimize the time to deliver value to the customer. The cost that customers are paying is realized very quickly. The cycle time from completion of a task or story to the production migration is significantly reduced.

- Core Activities of Business are focused more by the IT companies because DevOps allows them to manage other activities very efficiently. The team can focus more on the core business activities because deployment pipelines are automated, and obstacle within the value stream is removed. Rather than just moving bytes and bits, one can focus more on the creation of more customer value. The organization gets better outcomes in business and more advantage in the competition, which is sustainable with the help of activities of DevOps.

- **Continuous integration might reduce productivity**.
- In continuous integration, the product is made live after the first working model of the project is created. Then after additional features are added promptly. The priority of the project manager might be to launch a few new features to the project as well make sure their teamwork well enough to meet the deadline. But the problem is, the development process cannot be planned. There can be certain conditions where the developer had to stop and fix some software bugs which is not in the plan and might slow down the production process. Also, the developer may think to make extra efforts on unexpected error will not be appreciated. This can defeat the process of adaptation.
- To solve this
- Firstly, do daily stand-ups with all your members of the team and make them understand their role in the upcoming continuous integration.

- Team members about the cost and advantage of continuous development.
- Make a road map for developers that tell when and what the coders will be benefited by doing their work in full potential.
- Having CI into the existing development process
- When you migrate from your current development process to continuous integration methodology, there might be cases when the project requires to change some part of the development workflow. It’s not an easy task to change from one development process to the other. If you choose to modify the operation of the workflow to CI, you must take precautions before getting into the process of migrating; otherwise, it might hinder the development process productivity. An elegant and perfect plan should be created to migrate from one to the other methodology.

- **Adapt to the new way of testing**
- As in the case of continuous development, your team might be testing the project at every stage in which it can slow down the development process. Hence more tests will result in writing more test cases and testing them as well which consumes more time. Therefore the developer should decide their working between writing test cases and further fixing bugs. A developer might be tempted to test their build on the go to know any of the errors. But this should be done in a much more systematic way. Developers should create test cases on the go which can be used by the tester in the process of testing. Thus saves time for both examiner and developer.
- **To solve this:**
- Get a habit to write test cases from the start of the project. It could save time and cost for the team, which also leads to good test coverage to the project.
- Also, make your team know the fact that development

- **The error message shouldn’t be ignored**.
- Developers should not ignore error messages as error messages are meant to be read. They are thus giving developers some hints to solve those problems. Ignoring an error message is stupid enough that can cause wastage of money, time, resources and might lead to colossal rollback.
- Continuous Testing
- **Lack of Environments**
- There is a lack of environments, sometimes while implementing the principles of DevOps because continuous testing requires more testing by frequently hitting lots of situations. Many Environments are sometimes based on the APIs whose availability depends on the provider of the API.
- **Creation of Feedback Loops**
- One cannot conduct continuous testing if he does not receive feedback frequently. Visibility of test execution and results are equally important as the automation of constant testing.

- Overcoming the dev versus ops mentality
- In many organisations, we see the old cliché of developers tossing code over an imaginary wall to a centralised operations team—where developers are trying to *innovate* and *make changes* as quickly as possible, and the operations team are trying to *maintain* high service levels.
- The objectives of these two groups often counter each other, causing friction points and resulting in handovers and increased costs, along with longer feedback loops.
- Devops is all about integrating teams together and breaking down silos within IT organisations. This journey begins by setting out a vision on *how* this will work for your organisation.
- Understanding the roles and responsibilities of where dev stops and ops currently starts, and how these can best be integrated together, is a great starting point for any company, and it’s often the first hurdle that it needs to overcome as it adopts DevOps practices.

- Common understanding of Continuous Delivery practices
- Once you’ve identified that your code needs to be continuously delivered to minimise feedback loops, and your engineers have implemented pipelines and ci tooling that enable you to do this, you need to ask yourself this question:
- Do your feature teams *really* understand what it means to continuously deliver your software into your environments and at a greater frequency?
- Most organisations will have their own definition of what Continuous Delivery means for them. For us, we define Continuous delivery as a set of processes that allow you to reliably and sustainably release new software changes of all types (new features, bug fixes, etc.) by ensuring that your developers’ changes never break the main project—maintaining it in an always-deployable state.
- The process allows you to verify that your full project is in a workable, clean state, before deploying to the production environment.

- Moving from legacy infrastructure & architecture to micro services
- Older infrastructure and applications with complex architecture stacks can be problematic, even if they have served the company for years.
- Maintaining the status-quo can often lead to stability problems, lack of support and high operational costs—all ultimately resulting in being left behind the competition.
- Using infrastructure-as-code together with a micro services architecture is a huge step towards a future of continuous innovation, which results in directly re-inventing and modernising the entire software development lifecycle and allows the business to quickly adapt to changing markets and customer needs.
- Moving towards a more cloud-native ecosystem with micro services architecture can open up the floodgates to faster development and quicker innovation. In addition, it's vital to have a solid foundation around automation, configuration

- Implementing a test automation strategy
- Your organisation already knows that automated tests are really important and are a key enabler for DevOps practices including CI/CD. So what’s stopping the slow down on test automation
- It’s not just about saying what the test strategy is, but also going one step further with sample implementation of that strategy as a guiding north star for the teams.
- This includes things like BDD practices and the three-amigos approach, as well as answering key questions such as:
- How do we do data management for our tests?
- Can we use open sourced shared libraries and common practices?
- What does a good end-to-end test look like for our code base?
- What should our smoke tests really do?
- Having a clear understanding of how to implement the test strategy can go a long way in getting test automation adopted

- Too much focus on tools
- With the exciting prospect of adopting DevOps, flashy new tools in the market can seem like they solve every problem under the sun.
- However, with the introduction of new tools comes the need to train your staff on how to use them, ensuring they meet security requirements and are well-integrated with the existing infrastructure.
- All this can divert you from your key priority: *your team*.
- Your team and your org structure are key to DevOps. Once you have the correct structure in place, the processes of the team will follow. And once the processes are defined, then you can determine the tools required to meet the processes.
- The people on your team are the most important factor when transitioning to DevOps. If they’re not trained on the newly implemented processes and tools, there will be confusion, slowing down the adoption of DevOps practices.

- Team ownership for deployments and releases
- In organisations where DevOps practices are being implemented, we still see that teams do not have full ownership of their deployment and release cycles of their software.
- This is often due to lack of understanding of the difference between deploying and releasing.
- Deploying is your software being installed, so to speak, into any environment, including *dev*, *test* or *prod*.
- Releasing is then taking it one step forward, and making it available to the end-customer. This is important so that the full cross-functional team concept of “you build, you run it” can be implemented properly.
- A good way to go about this is for the team to start working closely with any ops folks and taking on shared responsibility for deployments, releases and operations, so there is shared context between the two. This allows devs, for instance, to empathise with ops teams on what it takes to actually deploy & release their code in production.
- Having this context can help in adopting DevOps practices for the entire team, and can enable the team to start taking ownership of not just deployments, but also releases!

- Resistance to change
- The move to DevOps can seem scary to some team members and key stakeholders. Packaging it as an *evolution* of current development practices rather than a *revolution* can help that issue.
- Telling people that they need to change can be seen as a bad reflection on the person receiving the advice. It should be emphasised that a DevOps transformation doesn’t happen overnight; it must be smooth and gradual. This allows everyone to embrace the DevOps culture as they slowly become accustomed to it and realise the different ways they can contribute to the development process.
- A good approach is to find a small product or full stack slice of an existing application to remodel it into DevOps practices.
- Once teams can see the benefits working in action, then other teams will organically want to adopt the new ways of working. This will steadily ease the sense of unfamiliarity, and get everyone on board to enter the new world of DevOps.

- <u>Key metrics are being acted upon</u>
- It is hard to argue against numbers and data and many organisations have started collecting all sorts of metrics. However, sometimes organisations focus on collecting too much!
- It can be easy to get pulled into a wormhole of metrics and dashboards. While these can be ventured upon with good spirits, it can quickly become a long and cumbersome process to collect basic data.
- It’s important to remember why metrics are being collected: so that they can be acted upon—and that’s the key, to make improvements so that those metrics improve.
- One way of doing this is by focusing on only collecting the dora metrics and making it available for teams to consume this, with agreed (and realistic!) actions on how to improve these metrics for those particular team(s). This focused and targeted approach can enable these teams to start adopting engineering practices that can then help to deliver and embed DevOps culture into the team.

- Dev and Ops toolset clashes
- It can also be an issue when dev and ops teams have completely separate toolsets and metrics. As simple as it may seem, it is beneficial to sit both teams down and seek to understand where it makes sense to integrate the tools they use, and unify the metrics they monitor.
- Some teams may be unwilling to part with legacy tools that are not only technologically inferior, but also slow down the entire infrastructure due to compatibility issues. Make sure the tools that are being implemented are aligned with the goals of the company and do not distract from your main objective.
- Overcoming these basic challenges in the beginning will make the move to DevOps much smoother. Over a period of time, every team member will get used to the feeling of constant change and innovation.
- Once the dev, ops and other teams learn to cooperate, they will determine ways of helping each other out, and collaborating even more closely.

- Getting started with continuous learning
- Curiosity can be a strong motivator for many to start learning! By far, one of the most important enablers for the team to start adopting a DevOps mind-set is the curiosity to continuously learn, adapt and improve their skills and knowledge.
- One way of achieving this is by ensuring that there is a platform that is available for teams to enable learning and sharing. This can be through communities of practices where the organisation invests in a day of learning and knowledge sharing once a month, or through lunch and learn sessions, or adopting *guilds* across teams.
- When new technologies or products are introduced, the concept of *dojos* can help coaches create a classroom environment for students to learn and upskill on key changes in the organisation, with a focus on hands-on practical learning.
- Similarly, a train-the-trainer model can also be implemented to ensure that practices are shared across all the teams, regardless of how big or small the organisation is. A culture of continuous learning, no matter which approach you take, can be the first small step that turns into a big leap in adopting DevOps practices.

ADVANTAGES OF AZURE

- Azure Cloud Services and Data Security
- Microsoft leverages decades of being a primary enterprise operating system and productivity solution provider to secure all information hosted in their Azure cloud products and services.
- An enormous amount of resources and effort go into securing an organization on-site data centers, where the slightest erroneous configuration can expose the company to advanced threats and exploits.
- With Microsoft Azure, all elements related to information security come included with the service, and customers gain access to the **latest in threat prevention and detection technologies.**

- Microsoft Azure Cloud Security Center
- As part of the Microsoft Azure Cloud Security Center, a range of security solutions and architecture implementations ensure all business data remains protected.
- These include **Advanced Threat Protection and Threat Intelligence** reports using Windows Defender analytics, and **Information Protection** that classifies and protects specific information from unauthorized disclosure by applying labels to documents and emails.
- Similarly, with network segmentation and encryption, every transmitted byte of data remains secure from any outside threat. Microsoft Azure Cloud Services uses threat monitoring agents that generate real-time security alerts and threat intelligence reports. These threat intelligence reports provide insights into attackers, their objectives, and the tactics employed.

- Improved Scalability and Business Sustainability
- One of the best advantages of Microsoft Azure Cloud Services comes from the scalability that the platform provides businesses.
- Traditionally, as companies expanded and grew, provisioning new infrastructure for them would be both a lengthy and a costly process for the organization.
- With a strategically deployed set of cloud center that span the globe, the Microsoft Azure Stack guarantees availability for services.
- In the last year, Azure averaged an uptime 99 across its cloud infrastructure—meaning **a total downtime of 26 minutes per year, worldwide.** Microsoft plans to improve this number by adding more servers to its regions by 2021.
- Currently, Azure Cloud Services utilize physical infrastructure distributed over every continent. This improves access times while also allowing for replication in cases where companies want to expand their business operations to outside the US.
- For data management systems, Microsoft Azure Cloud Services use elastic pools to ensure the same database performance levels that companies expect from their locally hosted systems. With elastic SQL database pools, organizations gain the best price-per-performance benefits while staying within their prescribed budgets.

- Simple and Effective Disaster Recovery
- Data loss presents a serious link to any organization. The impact of both lost revenue during the downtime, as well as the effort required to reconstitute the information can cost a company hunders , if a suitable, business continuity procedure isn’t already in place.
- A disaster recovery plan requires multiple built-in redundancies while achieving an acceptable time-to-recovery objective.
- With Microsoft Azure, companies can enforce specific recovery criteria for different BUs (Business Units) and types of data. Businesses can test their plans for efficacy as frequently as required, and find the perfect balance between the cost and control of their data recovery processes.
- Running backups and advanced replication of applications in a cloud environment will reduce the time required to recover from any type of outage, while also limiting the risk of becoming a victim of a ransomware attack.
- The Azure Site Recovery tool allows companies to create recovery plans that include replication, failover, and failback procedures from a single location. Azure Site Recovery also enables companies to **reserve IP addresses for applications, establish and configure load-balancers, and integrate Azure Traffic Manager for seamless network switchovers.**

- Advanced-Data Analytics and Business Insights
- The Microsoft Azure stack of technologies comes with built-in analytical solutions to shape and visualize data for improved decision making.
- Real-time dashboards that cover both customer response times, data and information flow and processing, and the ability to analyze a big data companies can improve business workflows more frequently and accurately.
- Advanced analytics can assist companies with segmenting customers based on behavior and ensure a personalized experience, further improving their satisfaction.
- Microsoft Azure Cloud Services include Machine Learning utilities with adaptable models to proof new approaches and analyze initiatives before releasing them to the market.
- For streaming data analytics, Stream Analytics enables organizations to set up a complete, end-to-end pipeline within a few minutes. By implementing the Cortana Intelligence Suite, companies can find ways to streamline their processes and transform data into intelligent actions.

- Integration Capabilities and Mobile Workforce Management
- Apart from the Azure Active Directory (AD) as an access management system, the platform also provides advanced integrations with third-party solutions via an API (Application Programming Interface) and direct integration with Office365.
- Additional solutions such as delivers a complete Enterprise Resource Planning (ERP) system to improve business workflows and increase oversight. As the Microsoft Azure platform is available from connected mobile devices, meaning SMBs are able to utilize IoT technology and improve productivity
- As most companies already rely on Windows-based software to execute the majority of their daily tasks, using the Microsoft Azure Cloud Services platform makes sense when moving to a digital-first business model.
- Compared to other cloud solutions such as AWS, an Azure implementation is easier and causes less disruption for most businesses. Making the transition to Microsoft cloud services is an easy and natural fit for businesses which have been familiar with their technology for decades.

- Regulatory and Legislative Compliance Frameworks
- A cloud implementation does require considerations relating to regulatory and legislative requirements.
- One of the advantages of Microsoft Azure Cloud Services is a shared responsibility relating to the information security policies and compliance frameworks. As Microsoft provides the data security tools and systems, the only responsibility that resides with the company is to classify and control specific types of data.
- Moving to a cloud-based digital technology stack will require companies to evaluate the solutions against their in-house information security policies.
- Azure maintains a compliance portfolio to ease the burden of regulatory requirements for their customers. With the Microsoft Trust Centre, companies can access compliance guides and setup audit reports for their data and information, ensuring they remain within the frameworks regulating their daily business activities.

- Azure Cloud Services and Data Security
- Microsoft Azure Cloud Security Center
- Improved Scalability and Business Sustainability
- Simple and Effective Disaster Recovery
- Advanced-Data Analytics and Business Insights
- Integration Capabilities and Mobile Workforce Management