One of my biggest and very recent achievement that I would like to highlight today is - strengthening Automation practices and solutions for CICTCD in Bosch across divisions.

Being part of the central unit in Bosch Germany for Mobility sector, my division drives and enables SW innovations and initiatives in an urge to make Bosch a SW or IoT company.

As we are evolving in vehicle E/E architecture and moving towards centralized Vehicle computer platforms, importance of SW in mobility has become inevitable. The future looks like all connected with integrated cloud and edge technologies and disrupted market demands swiftness – all in all this calls for a Continuous development and deployment.

My case study revolves around the same, when we as a team brought management attention towards CICTCD in SW and product life cycle but as we progressed, there were major roadblocks. I then took up a challenge of one of such roadblock and finally created a big opportunity out of it.

Let me take you through my Journey in Autonomous program. And this is the case study I would like to present today as my achievement which not only demonstrates just one success but it’s an amalgamation of successes at each level using most imp levers called collaboration and co-creation.

I have structured the case study in such a way where I would take you through what were the initial problems, how I structured those problems and provided a strategic direction. How the gaps were assessed and how I laid down the foundation of Automation platform not by just creating new solutions but mainly by introducing the culture of Re-usability - Proof of Scale rather than Proof of Concept. I would also show how this framework has already started benefitting organizations amidst all the challenges related to Team, Stakeholders, Interest & Motivation.

I started this journey by understanding the status quo in projects. The expectation was clear – fast, furious and automated pipeline. So, I used process metrics to find out ‘ where we are and how fast we are’

This model of process gaps was an eye opener for me. I used this and pilot it in some projects but overall analysis remained more or less same – Even though processing time is 10 days, lead time is 70+ day with huge delay. Various reasons but boiling down to the same problem – too much of waiting time, many manual steps, late feedbacks, release cycles not in sync, and even if Automation solutions are available they are not catering to the need of value chain.

As a result, I volunteered myself to enhance the Automation in overall SW life cycle and after getting knowledge of the real time processes and gaps from the divisions, I crafted a simple model called 3C model that I have used for this transformation and Change management - Collaboration, Co-innovation and Competency.

Collaborated with multi-functional and domain teams for bringing synergies and agreements on common technological solutions, and closely collaborated with other on-going initiatives like Hosting platform so that we just take benefit of utilizing the expert’s craftsmanship.

Co-innovated solutions through tribe of experts in creating domain agnostic solutions.

Competency enhancement by providing right platform for many aspiring people who wanted to break their mechanical routine and wanted to learn new-age technologies. Education programs were crafted to make strategists, IoT and AI experts of such team members.

Now following the 3C model and also accounting other leadership aspects like Change management, Stakeholder management, Comms – which had to be injected and tracked all along the program, I came up with high level plan and segmented it into Vision, Landscape, Light House projects and Collaboration. Thanks to PMP and my project management experience as If I would have jumped directly to Technical solutions, which ofcourse is the key, it would have been difficult to penetrate and get acknowledgement/commitments from all divisions.

I will be briefly covering all the segments just to give a glimpse on how I executed the entire plan.

I created a vision of Autonomous Testing to cover breadth and the depth, transformation to Autonomous by 2025 . Took required buy-ins, approvals and budget from Stakeholders consisting of Unit heads, which had its own challenges. Once I had green signal to convert this idea to full fledge program, I approached delivery leads to lend me the SME’s who had shown interest & trust in Autonomous concept.

And there I was welcomed with another challenge of non-availability of some imp SMEs. As I did some trade offs to get the attention of Delivery leads for their support and get their experts available for this program. In-parallel I was lucky to get Student support from Germany who was allocated to me as part of mentoring program. Together with her, I created questionnaire to access the maturity of Automation in Continuous Integration, Delivery and Deployment categorizing in 5 areas of Build, Testing, culture, deployment, and release.

This questionnaire is now a full fledge tool which is being used to assess the maturity of automation in CX pipeline. This came as small but very encouraging achievement. I would be honest, even this process underwent it’s own challenges as penetrating such practice of filling 100+ questions to just know the status – for developers, leads as we expected, it was merely a waste of time.

Let me move to next swim lane of Solution Landscape wherein I gathered Product champions (one per Product line) and created a repository of available solutions. The numbers that came from this collection exercise were surprising and alarming. This risk had already been identified earlier and as part of mitigation plan, qualification gate was already ready. This was another small deliverable from our collaboration activities with Product champion and together criteria were identified to evaluate the solutions.

The slide that I am flashing on my screen is just one of the example where we automated Defect management pipeline to get feedbacks and anomalies right at the time of coding.

1. This is a logical approach for an intelligent Defect Management process. Where using Binary class prediction and source code, anomalies are predicted in the code. A great tool for developers to ensure built-in quality.

Feature Selection based on Correlation analysis, Voting Selector & K-best Features

Addressing Class Imbalance using SMOTE, ADASYN & ROS , GAN

1. Irregularities found in code can be a signal for error-prone feature/requirement which indicates strategy for risk based testing. This would enable early feedback to developers.
2. Using deep learning algorithm (& ML based representation), defect resolution approach can be localized in the source code and the same can be prescribed for quick fixes.

This is one example of newly produced solution based on need from divisions.

Let me take you to another example which leveraged 2-3 available solutions and just by integrating the existing solutions, the end to end value chain was optimized. This is one solution which is implemented in one of the IoT project in Home Appliance business division and reaping good benefits.

Using MQTT, the integration flow is made with HMI and as soon as Mobile apps are ready, it can be integrated and validated using message protocol. Similarly, with ever growing need of third party interface - automation framework was created to test voice enabled devices and business flows.

To summarize, this entire exercise I was working for four targets under the umbrella of Automation & AI solutions:

Prood of scale, Integration of solutions, New Bets, Outside-In

1. Proof of scale – Stop reinventing the wheel and re-use with minimal customization.
2. Integration of solutions – Check the possibility of integrating some solutions and optimize the value chain.
3. New solutions – Gaps in covering the breadth and adding AI flavor to it. Make it more intelligent.
4. Outside-in perspective – Make vs Buy options.

Quick glimpse on machine based code reviews & static analysis.

Left side solution is in house-built and identify source code change impact from daily builds and provides early feedback for developers to adjust their code. Right side is from deepcode.AI. Rather than developing all on our own, we also plan to take some built-in expertise from outside. This is one of the example – the tool which is different from SonarQube, QAC and is a textual analyzer, doesn’t build the code. It can be integrated with bitbucket pull request workflow.

Before I move to next segment – which shows the successful implementation of this framework. I would also like to share some challenges in this roller coaster journey, which indeed turned out to be greatest learning & experience.

Carrying a strong experience as a Line manager with direct reports, this experience of managing dotted line was quite different, it had it’s own rewarding challenges . It becomes even tough if allocation is on volunteer basis especially for senior experts or if the allocation is not 100%. I’ve had the feel of all such scenarios.

Another big challenge was ‘fear’ of losing ownership, porting project data to new tools, management fear of demotivated team by introducing automation & AI.

However the underlying issue is change management as this framework has an impact on all levels and their way of working.

Inspite of these challenges arising in the project life cycle I made sure most of these were dealt with tactfully, while others are being sorted out as the program is still continuing.

Moving to next segment – the framework of entire assessment, analysis, consult & implementation was then piloted in 2 product lines – I picked one from BBM and other Non-BBM.

When we started this program, it was never meant for Non\_BBM. However it was not even restricted to only BBM. Thanks to wide Stakeholder list and the nomination for the implementation was received from home appliance division of Bosch. Process gaps I have shown in my first few slides was for this connected product with Lead time of 70+ days.

One of the major reason being release cycles not in sync with each other but on the other hand, with many silos, the overall release cycle was way too long. Manufacturing of the appliances were taken care by I4.0 and was best in class. Mobile application was from other division of Bosch. Cloud for data processing happens in AToS and for scaling AWS is being used.

As we worked together, our focus was on end to end connectivity where the major issues were found. As part of Autonomous strategy for connected devices, we extended defect prediction with inputs from Market Data, Crowd data, Support tickets.

I implemented integrated solutions i.e MQTT to get message protocol from hardware. All-in all I was trying to break the silos and shorten the entire process cycle resulting in reduction of lead time from 71 days to 28 days.

This resulted in fast Go to market and turned out to be the biggest achievement as Bosch home appliances were in high demand in 2020 and this division was able to successfully satisfy the market need.

And I think this framework and program has already given due benefits to not only me as I had multiplied my leadership and technical skills but also to organization as it helped in reducing process cycle and thus enabling faster-go-to market,

Built-in Quality has definitely improved due to activation of various fast & constant feedback mechanisms.

re-usability has been increased, that means development cost of new solutions are not required. It also means that the maintenance of existing tools/solutions are not needed. This way there is huge cost savings realization. A factor of it was shown in earlier slide representing piloting.

I would like to end here with my envision for this program with two cents -

1. Moving from Talyristic approach to Full stack SW Team.

Each handover from one team to another team is error prone. Even when everything is implemented precisely according to the specification, there is a significant risk that the whole feature chain does not work as intended\*.

On other side Full stack team which I also refer as jack of all trades. Within the boundaries of its domain specific expertise, a Full Stack SW Team is capable to develop a shippable increment including V-model artefacts with little or no support from other teams.

* Projects split in E2E feature teams indicate high CX maturity
* Teams work with a clear definition of done on the whole V
* Competence for the left and right part of the V in the team becomes important

1. Having an Automation den.

Given the authority, money and approval – I would like to take this program one level above and create a suite with marketplace or app store kind of feature. Users who are consumers i.e developers should be able to search the needed automation solutions and create their own workflow. I would like to promote social coding culture and re-usability to its best.

Additional thing sometime

Why do I take extra work – by motivating them off them opp and exposure, learnings, apperecition in various forums, they have received accolades.

Showing business case and success story

Guaranteed year on year return

We take ownership of centrally managing tool.

Power central challenge.

India vs Germany