How behavioral change fuels DevOps

Effective method for behavioral change enables the transformation to the DevOps way of working

All over the world organizations have embraced the DevOps theme, or are at least considering it. The transformation towards this new way of multidisciplinary collaboration should yield increased agility, speed and customer satisfaction. Behavioral change is the catalyst here. But what approach for behavioral change is most effective?

The fact that many organizations, also in the Netherlands, have adopted an Agile way of working is nothing new. That doesn't mean that these organizations actually are continuously and rapidly deploying new or changed software. Especially 'the last mile' seems to be the hard part in the production process, causing organizations to stumble before the value is actually being delivered. DevOps aims at an organizational mindset for continual improvement of the digital value chain, by enabling cross-functional collaboration on the levels of process, technology and people (behavior).

The DevOps movement mainly originates from start-ups and high-performance organizations, of which Flickr, Amazon, Netflix, Facebook and Etsy are known to be highly successful. Obviously this has to do with their innovative culture and their business dynamics. In the last few years DevOps has become popular in many other environments as well, as we can see in the Netherlands. Banks, insurance companies, logistic service providers, industrial multinationals, energy companies, but also (local) governments are currently taking the first steps in the field of DevOps. Some companies choose to make rigorous changes and have completely transformed their organization towards a DevOps collaboration model, others are still in the experimental stage.

Context

The broad attention for DevOps is directly related to the exponential growth of the Lean Start-up movement. This is no surprise, as famous start-up companies like Netflix and Spotify have propelled the DevOps way of thinking. Next to that digital disruption is a sign of the times. This requires and draws special attention to the digital production process as well.

Traditional organizations want – no, have – to change as well. In order to keep up with the modern start-ups and their dynamic markets, many large multinationals launch programs aimed at organizational and cultural change. Modernizing organizational culture, people and resources / assets usually yields both cheers and resistance. Specifically when people have been working in their silos (design, development, testing, operations) for years, we see behavioral patterns reflecting these silos. Breaking and effectively changing these patterns is one of the most critical success factors in any DevOps journey.

This is related to 'Conway's Law': organizations that develop systems are limited to producing designs that reflect their internal communication structures. In other words, if we want agile and

resilient information systems that facilitate the entire “customer journey”, the whole organization (including culture, behavioural and communcation patterns), has to make the same shift.

DevOps results

DevOps is mainly linked to speed, more specifically to higher speed of deployment towards production environments. However, there are more results to expect from the DevOps way of working. The most interesting results we see are:

- Improved customer satisfaction:
  The customer (‘business’) is actively involved in the process, usually in the role of Product Owner. Teams in the DevOps context not only deliver functionality, but also create the conditions in which the functionality can be applied most effectively.
- Value – Risk balance:
  The DevOps mindset improves resilience for both the organization and the systems used. Resilience is created because the organization allows itself to explicitly learn from failures and setbacks. Risks are mitigated by reducing the batch sizes and by enabling continuous interaction (feedback) between all team members and stakeholders.
- Reduced time-to-market:
  Reducing the units of work and automating steps like integration, testing and deployment lead to faster implementations. Next to that, early stage experiments (e.g. hypotheses and prototyping), but also sharing knowledge and work among developers and support engineers before and after go-live enables continuous optimization of feedback loops.
- Cost efficiency:
  DevOps organizations differentiate themselves by explicitly applying the concepts of multidisciplinary teams and cross-functional behavior between team members. E.g. taking over (even partly) each others tasks, or consistently sharing knowledge, practices and templates. Similar behavior is also identified between teams, which leads to new communities, increased reuse and sharing of knowledge and experience in continuous delivery pipelines.

CALMS

DevOps is so much more than just a set of continuous integration and development tools. Proof of that can be found in the so called ‘CALMS pillars’ within the DevOps community. A keen balance between those pillars is crucial for an effective adoption of the DevOps mindset in every organization. Too much focus on automation will lead to an organization which has a state of the art toolset, but which lacks collaboration, empathy and the capability to improve. The CALMS pillars are:

- Culture: a DevOps organization kan automate all they want, but as long as the professionals and managers do not have the proper competences and capabilities and do not show the required behavior, the silos will never be broken down, or even connected. Cultural change

requires an integrated approach, which - in small increments – builds an integrated and continuously learning organization.

- Automation: In DevOps environments it is common practice to standardize and automate as many manual activities as possible. Tools are an important catalyst for realizing continuous integration, consistent version control and configuration management, automating deployments, monitoring and provisioning.
- Lean: The DevOps way of working uses elements from Lean, Agile and Theory of Constraints. Lean elements such as Flow, Pull and Kaizen are used in all processes, as are incremental improvements on bottlenecks ('ToC'). Value is optimized, waste is minimized and flaws and defects are resolved at the source as much as possible ('no defects downstream').
- Metrics: in order to be able to improve, reliable performance information is a must. Professionals and managers use metrics, such as the number of deployments per timeframe or team member, or the number of improvements realized, in order to manage and improve performance even further.
- Sharing: DevOps teams contain a mix of roles and competences, which preferably all work at the same location, using adequate knowledge management tools. Sharing knowledge regarding requirements, development, testing, maintenance and other relevant roles, materializes the concept of continuous streams of feedback in the multidisciplinary teams.

Focus on Culture and Metrics

In this article we focus on two crucial elements from the CALMS pillars, being Culture and Metrics. These two CALMS pillars have a tight relationship and a strong influence when it comes to succesfully applying the DevOps way of working.

Culture

Culture can be seen as the way people in an entity (team, group, department, organization, country etc.) interact with each other. Written and unwritten 'game rules' are translated and applied within a certain context; "this is how we do things around here". Culture reflects the habits of an organization. When DevOps is introduced, this will have effects on the culture, the habits and therefore on the behavioral patterns within the organization.

Because DevOps affects and changes culture, there is an important role for the leaders and managers in the organization. They have to explain what DevOps means in terms of culture and cultural change. Simply put: the rule set for behavior will change. Some behaviors need to be adapted or others might even be eliminated. New behavioral rules are introduced. Rules are always tested on scope and consequences. Behaviors which are allowed and reinforced become habits. This is true for both desired and undesired behavior. Slowly but surely a new set of functional and less functional behavioral patterns emerges and becomes the new way of working; the new culture.

The behavior you see in organizations is conditioned; people have learned which behaviors are required and allowed and which are not. When you are new in an organization, you quickly learn which behaviors are required to become an accepted member of the group. For instance is

everybody wearing 'shirt and tie' or are casual clothes more acceptable. Displaying proper behavior leads to all kinds of reinforcement (e.g. compliments, being accepted as a group member) for the performer. Reinforcement leads to an increase in behavior. When a group member displays undesired behavior, this will lead to negative responses (negative consequences) from the peers. Someone[1] who just joined a team and outperforms the others in the team (and the team considers that to be undesired behavior) will be ignored during coffee and lunch breaks. Only when the new member lowers her tempo and comes in sync with the other group members, the behavior of the group towards her changes and she receives more positive social consequences from her group peers. This example shows that reinforcement mechanisms can also lead to undesired performance levels.

Metrics

Management is accountable for the end results (outcomes) of an organization. Triggered by the need to manage progress and development of 'the numbers', the management selects a number of (performance) indicators to be able to manage their organization.

Metrics used by (project) managers have a very clear relationship with the behavior and culture in an organization. The relation is that "you get what you reinforce". In other words: you get exactly that kind of behavior that you have invoked in some way by managing on certain metrics. The metrics you have selected, result in certain behaviors; people are inclined to achieve the results. This can lead to both desired and undesired behavior. When you observe undesired behavior, this behavior is somehow, perhaps even unintentionally, reinforced.
This mechanism is crucial when implementing the DevOps way of working. As a result of the DevOps transformation the metrics and reinforcement mechanisms change which leads to changed rules for behavior. Therefore it is highly recommended to select metrics that will lead to desired, value adding behavior.

Professionals are inclined to make a difference, add value to your organization. They also need to have something in return that is valuable to them, e.g. recognition from the management for the results achieved. In many organizations we have noticed, that people doing exactly what is required, are in fact ignored. All management attention flows to those people who are underperforming. The reason for that is plain and simple: underperformance is an opportunity for improvement! And it's the accountability of management to make sure that all goes well and that you improve where you can.

From culture to behavior

Now, the million dollar question is: how do we obtain this "desired culture", in which we see true "DevOps behavior"?

[1] Performance Management – A. C Daniels & J. S Bailey – page 109

A common approach for cultural change is letting the (senior) management send a lot of signals towards the people in the organization. The message seems crystal clear: customer focus, become more proactive and improve communications. Usually this message is supported by identifying and communicating a number of corporate core values, such as 'open', 'honest 'and 'collaborative'.

This approach contains some serious pitfalls: the messages and core values are way too generic for most people involved. These messages usually have little or no effect on the behavior of people on the work floor; there where the actual value is created. A simple reason for this is that it is not clear exactly which behaviors must be changed, stopped, or increased to actually have 'customer focus'. When in doubt, try to show the behavior to someone else. It is likely that you have to show a lot of different behaviors, if you want people to conclude that you are showing 'customer focus behavior'.

From metrics to improvement

Our view on the matter is that culture is an end result of all behaviors and behavior rules in an organization. Trying to manage by only focusing on the end result is not practical. It is like trying to improve service availability by only accurately measuring availability. It is far more practical to identify, monitor and measure all factors that influence service availability and improve where necessary. These improvements will eventually improve service availability in a more sustainable manner.

Achieving cultural change in this respect is no different. In a DevOps environment specific results have to be delivered in specific time frames and speed is of the utmost importance. Performance must be managed, hence metrics play an important role. Clearly pinpointed results, behaviors and relevant metrics are fundamental to any DevOps transformation.

Employees will focus their behavior on established metrics and try to deliver the desired performance. When employees have a good score on these metrics they might receive additional recognition and praise from higher management. That is why it is so important, with the DevOps way of working in mind, to be very selective and precise in choosing which performance and performance indicators you want. Core question is: which variables should we measure in order to build DevOps behavior and sustain a DevOps culture? See <box 1> for some examples.

Since a few years a rapidly growing number of organizations are using 'Organizational Behavior Management', or OBM. This is a concrete approach, focused on performance improvement, which has its roots in behaviorism and behavior analysis (Skinner, Thorndike, Sulzer-Azaroff et al). The goal is to create an environment for employees in which they *want to perform*.

The power of OBM lies in a simple and effective protocol that leads to sustainable performance improvement and behavioral change. Essentially the protocol is all about pinpointing, measuring and reinforcing desired behavior in order to achieve the desired results.

This protocol is also applicable while introducing the DevOps way of working. Basically, the DevOps way of working will only be successfully adopted when everyone involved continuously shows 'the right behavior'. When all involved actually join in and commit, then new behavioral patterns and a new culture emerges. The new patterns have to become habitual and OBM is an effective method to do just that.

Seven Steps

The OBM protocol for performance improvement and behavioral change consists of seven steps:

1. Pinpoint
   The protocol starts with pinpointing (specifying) the desired performance. Performance consists of two elements: a concrete result and the behavior that will lead to that result. When both are adequately pinpointed it becomes very clear which are the desired results and behaviors; there is no more room for ambiguity. It's very important to understand that adequately pinpointed behavior is measurable. This enables you as a manager to effectively manage performance.
2. Measure
   We measure the frequency of the desired behavior. When you first measure behavior over a period of time, you inherently create a baseline. The relevance of the baseline is that it enables you to see any change in performance, when you compare against the baseline. It is crucial to use the measurements to reinforce any performance improvement you see, no matter how small. From a management perspective it may be tempting to use the measurements as an argument to present negative consequences to the performers, when performance targets have not been met yet.
3. Analyze
   The next step is to analyze the behavior that leads to the current performance levels. We use a behavioral law stating that behavior is controlled by its consequences. By analyzing the consequences of behavior, we can see what we need to do to either increase or decrease the frequency of that behavior. While doing that, always take the performer's perspective. If you want to change behavior, you must change the consequences for that behavior.
4. Feedback
   Any team needs concrete and accurate performance feedback. Performance feedback shows you where you are now, in relation to a goal. The best way to present performance feedback is to make a graph. Keep it simple, just use a whiteboard or flip-over and make sure the team can easily see the graph and any progress made towards the goal. Performance feedback informs the team of which behaviors should be changed or increased to achieve the desired results. Introducing[2] proper performance feedback will quickly lead to significant performance improvement.

[2] Rantz & Van Houten (2011), Journal of applied behavior analysis

5. Setting (sub)goals
   Goals need to be concrete, attainable and challenging. The end goal is divided in a number of smaller (SMART) goals. This prevents the use of so-called 'stretch goals', which will only raise doubts regarding the attainability of such goal. Doubt and fear of failure have a negative impact on both motivation and performance of all involved. The other way around is also true: being successful in reaching a goal, no matter how small, will spark motivation!
6. Reinforce
   Attaining (sub) goals is rewarding. We reward attaining goals and we reinforce the behavior that leads to goal attainment. To strengthen proper behavior over time, a consistent mechanism for delivering reinforcement is required. We can use measurements to indicate how far a team is from reaching the goal. Goal attainment (and the measurements are proof of that) should lead to celebration of the achievement.
   A common issue with reinforcement is that money is perceived as the one and only way to reinforce behavior. Of course, money is important, but usually more money does not secure a consistent and sustainable higher level of performance. This is certainly true for knowledge workers. Effective ways to reinforce behavior are positive management attention, symbols (trophees or certificates) and extra time off. These are known to have much more positive effects on performance than just money.
7. Evaluate
   What went particularly well and what could be improved? Practice shows that adequately pinpointing performance hardly goes right the first time. Finding reinforcers and correctly applying them is also harder than it might seem at first. These areas have pitfalls and this is probably where you can find room for further improvement.

OBM & DevOps

OBM is a valuable instrument for realizing and sustaining behavioral change in DevOps environments. DevOps is most of all a mindset and culture driven organizational theme. Of course, technological possibilities, such as continuous integration and deployment, automated testing, micro-services, containers and cloud provisioning, are extremely important. Even processes evolve towards Lean and Agile to improve collaboration and transparency. But the transformation to a new way of working, mindset and new behaviors remains key. By shifting attention towards the required behavioral change, the DevOps transformation leads to more viable and sustainable results. The new mindset and behaviors are the basis for continual improvement and improved agility. DevOps is just the next step in the evolution of organizations. It can never be the end goal. That's for sure.

Some extra info for you to consider

****

Examples of DevOps behavior

The following list contains some examples which we have experienced to be desired behaviors in DevOps environments:

- Ownership: the Product Owner leads the demos and adequately informs all his/her stakeholders regarding progress and deliverables.
- Prioritization: The Product Owner and Team Members discuss at least once every sprint the priorities regarding both functional and non-functional work.
- Multidisciplinary: Team Members actively take over eachothers tasks when they see a colleague is either too busy or unavailable.
- Sharing: All Team Members personally share their experiences and knowledge amongst their fellow Team Members. They immediately update the wiki accordingly.
- Fail safe: All managers openly reinforce identification and resolution of errors and actively inquire for lessons learned.
- Empathy: All Team Members, including Product Owner and Scrum Master, openly discuss their worries and needs for support.
- Improvement: every high-impact incident is to be followed by a blameless postmortem, in which all members freely share ideas and improvement opportunities.

Pitfalls in behavioral change

A few pitfalls to avoid when working on behavioral change:

1. Only sending the message, assuming behavior will change.
   Behavior only changes when consequences for performers change. Telling someone what to do is not a consequence. Of course, communicating the reason for change ('sense of urgency') supports the change process, but communications alone will not guarantee any true behavioral change.
2. Only using negative consequences, such as punishment, to drive behavioral change.
   Of course negative consequences have effect on behavior. But punishment never increases

performance. On the contrary, it generates fear and causes performance drops, even in areas that used to perform well. Positive reinforcement works much better for everybody involved.

3. Reinforcement is commonly associated with *paying* more for *results*. Even though money is important for people, it is rarely a 'true driver'. So called social reinforcements for desired behaviors are way more important. Think in terms of compliments, attention and acknowledgement. The only condition to make this work is that the reinforcer always follows behavior. Otherwise it will only have a temporary effect, at best.

Dave van Herpen

Dave van Herpen is Independent Consultant, Coach, Trainer | Agile & DevOps transformation at IMP/ACT. Dave, in his role as advisor, coach, trainer and change agent, supports his customers to organize for end-to-end agility.

Robert den Broeder

Robert den Broeder is owner of Trigono BV and is a certified trainer / coach in Service Management and Organizational Behavior Management (OBM). He is the author of the book 'OBM, an introduction' (available in Dutch and English), which serves as a study guide for all official, accredited OBM Foundation level trainings. Robert supports organizations with implementing and improving service management processes, with special attention for behavioral change.