

2020-09-05

# Analytics-driven approach to agile software product delivery

---

Boncheva, Bogdana and Ivanov, Penko. 2020. Analytics-driven approach to agile software product delivery. Computer Science and Education in Computer Science 16th Annual International Conference CSECS 2020, Sept. 5th 2020, online, ISSN 2603-4794

<https://hdl.handle.net/2144/43062>

*Downloaded from DSpace Repository, DSpace Institution's institutional repository*

# Analytics-driven approach to agile software product delivery

Bogdana Boncheva, MSc, PMP, Certified SAFe 5 Agilist  
*Senior agile project manager*  
*Financial Times*  
Sofia, Bulgaria  
[bogdanaboncheva@gmail.com](mailto:bogdanaboncheva@gmail.com)

Penko Ivanov MSc, PMP, CBAP, PMI-PBA, PhD Candidate  
*Senior business analyst*  
*Financial Times*  
Sofia, Bulgaria  
[penko.ivanov@gmail.com](mailto:penko.ivanov@gmail.com)

**Abstract**—Two key factors drive the software product delivery - the ideas for new products, and the latest approaches for optimized development. This paper focuses on the software development process and shows how data analytics enable innovation and efficiency in the delivery of a new product.

The authors recommend the tools and techniques they have tested and proved successful in an international product organization within one of the leading media companies in the world. The presented analysis addresses the challenges of the standard practices in agile software development - continuous incremental product delivery and integration. This iterative approach implies developing and delivering features before a product, or even a product vision, are entirely complete. The method gains continuous feedback from the customer and adjusted revenue projections from the organization. The success of the approach relies on frequent and prompt decision-making by stakeholders from various backgrounds and with different skill sets.

These decisions need to be well-informed as they drive rapid changes in the work prioritization and scope, and in the focus of the software development team—those frequent shifts in direction impact the delivery time and the quality of the product. Decisions on affecting the different elements of the engineering teams' effectiveness rely on cumulative information about the teams' capacity, lead time and throughput.

This paper showcases how data analytics can drive prompt decisions and enable the necessary flexibility and improved efficiency. The authors demonstrate adapting the data visualization to the different audiences according to their interests and levels of expertise: customers, senior management, engineering teams. The paper advises how to choose the right data sets and make the correct assumptions for the data interpretation. The authors' extensive practice shows these are the prerequisites to making the right decisions and delivering the impactful products that make an organization stand out.

**Keywords**—*Software development process, Software product delivery, Agile software development, Data-driven decision-making, Data analytics, Data visualization*

## I. INTRODUCTION

Ideas for new, revolutionary products skyrocket companies to business and historical success creating new industries and pushing technology forward. Product innovation is guided by thoroughly researched and analyzed metrics and datasets. Market research metrics drive product strategy, and user research metrics drive design. These are metrics updated in real-time that elevate the processes of product discovery and product ideation to the highest possible quality level for the organization.

The delivery of the new products, however, is often guided by metrics that are not informative for possible improvements and efficiencies. Such metrics reflect the progress against schedule, budget, or scope. Those metrics provide transparency to stakeholders and help manage expectations. Analyzing earned value, cost, and schedule performance indicators do little to help delivery teams perform better or deliver to the highest possible quality level for the organization.

## II. PROBLEM STATEMENT

A major shortcoming of traditional project metrics is that the data driving the product delivery is no longer applicable midway through a project. It is based on a significant amount of assumptions that may be proved wrong during developing and delivering the new product.

Another substantial downside is that the data is benchmarked against schedule, budget, and scope in metrics only understandable by Senior Management and too theoretical for the engineering teams or the Customer organization.

That data is not aligned with the specifics of the delivery teams that are creating the new products. Analytics of such data sets do not lead to identifying improvements in delivery processes and engineering practices.

The data that is guiding product development needs to be: regularly updated, adapted, based on valid assumptions, correctly interpreted and enabling improvement ideas by management, by the teams developing the product and by the users that will derive value from that new product.[1]

This paper proposes data analytics that meets these requirements driving right agile delivery of a desirable, feasible, viable, marketable, and competitive product.

## III. LIMITATIONS OF THE STUDY

The following limitations have been identified and considered for the study:

- The presented data has been gathered for a period of two years (2018 - 2020).
- The engineering teams that have been guided by the presented data analytics are a combination of distributed, collocated and 3rd party teams with a joint program goal, delivering different capabilities and functionalities of one software platform, powering the company's core business.
- The team size has varied from 6 to 22 engineers.

- The analytics-driven approach was applied to leading the delivery of both a significant organizational change and new product developments.
- The organizations of the engineering teams are all using agile methods of software delivery: Scrum, Scrumban, DevOps, Pair Programming, Lean-agile practices.

#### IV. METHODOLOGY AND BACKGROUND

The methodology applied involves both major types of research:

##### A. Quantitative:

- Predictive and prescriptive analytics for Velocity/Scope/Capacity.
- Descriptive analytics for incident reporting, escaped bugs reporting, Achieved Outcomes.
- Diagnostic analytics for Blocked/Dependency work.

and

##### B. Qualitative:

- The motivation for improvement.
- Employee engagement.
- Engineering expertise.

The analyzed data has been gathered from the following tools that are used for registering work items: "Jira, Trello, Heimdall (the FT's monitoring aggregator), Response (a slack app originally developed by Monzo and used at the FT for managing and tracking operational incidents), and Roadmunk.

This research focuses on teams working in agile organizations, using lean-agile practices that empower teams to be self-organizing, highly collaborative, and able to change direction quickly.

The essence of the agile mindset is captured in the Manifesto for Agile Software Development:[2]

*People and interactions over processes and tools.*

*Working software over comprehensive documentation.*

*Customer relations over contract negotiations.*

*Responding to change over following a plan.*

Product development in an agile context is iterative, meaning that each successive version of the product is usable, and each builds upon the previous version by adding user-visible functionality.[3]

Agile projects are iterative insofar as they intentionally allow for "repeating" software development activities, and for potentially "revisiting" the same work products. They are iterative in a third, less essential sense, in being most often structured around a series of iterations of fixed calendar length.[4]

Another essential feature of agile product delivery is the multiple demonstrations of product to stakeholders and users long before even the first version is released for use. The goal of this rigorous demo process is to gather as much feedback as possible. This feedback is continuously actioned and may lead to changing the functionalities or design of the new product multiple times. Thus, when the product is finally delivered, it satisfies the latest users' and the market's demands and needs.

The product complexity and the team specifics often influence the length of each iteration. If the development is a collaborative effort between teams, it is highly effective for the iterations of the teams to be of the same length and fall in the same periods to achieve alignment. Adopting regular cadence of teams is a key success factor to maximizing collaborative potential, reaching synchronized flow and common goals.

The dynamics of the product delivery processes need to be reflected in the data analytics that guides the development efforts. The process in Fig. 1 was followed for creating the analytics-driven approach to delivering innovative solutions through the Content and Metadata platform of a world-class media company. The process assumes regular evaluation of the input to the different process steps and requires constant adaptation.

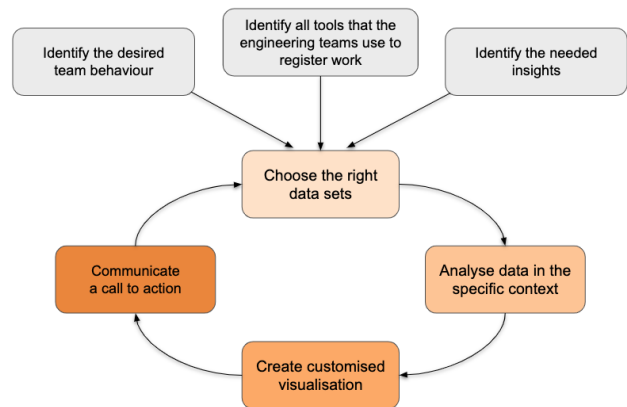


Fig. 1. Essential Analytics-Driven Delivery Reporting process

Delivery analytics have a strong influence on the way engineering teams perform. It is important only to choose data that will bring a positive impact on the values and behaviors in the team. Building an entirely new product, the concept for which is still subject to change, often requires complex technical solutions and challenging integrations. Breaking down work items to executable tasks should be guided by the values of clarity and quality. Evaluating team performance by monitoring throughput of work tickets may lead to breaking down work to unnecessary granularity and slow down the delivery of the actual product.

The tools, from which data has been collected for this study, have been selected to be the tools used by the development teams to log work. There have been very successful studies that have used code repositories for the data source of the delivery analytics.[5] The goal of this study is to use analytics to influence the organizational processes of the teams as well as the engineering practices. That is what makes it essential to select only the data that will bring teams' focus on delivering working software and not solely on completing work tasks.

Depending on the products' and the engineering teams' specifics, different insights are needed for the successful completion of a new product. Often, the insights required for prompt decision making change throughout the initiative, and need to be regularly evaluated. Such insights may be external/internal dependencies, capacity limitations, performance trends, delivered value, quality trends, technical

reliability, and uptime, etc. The delivery analytics needs to uncover those insights and to enable informed decisions.

There are many different roles in executing a product vision: Product Managers, Business Analysts, Principal Engineers, Senior and C-level Managers, Marketing Management, Business unit owners, Customer Program Management, direct users, 3rd Party partners, software development and platform teams, etc. Stakeholders in different roles possess different expertise and are interested in various elements of the delivery process. The delivery analytics needs to consider those differences and be customized for each type of audience.

The terminology also needs to be carefully curated to the professions of the different audience members. For delivery analytics to be effective, it is vital to consider visualization that is easy to perceive, clear to understand, unambiguous in the conclusions that a professional would draw from it. Using two to three visualizations that complement each other and deliver a unified message has proved efficient in informing decisions that were confirmed to be correct and timely upon review of already delivered products.

Fig. 2 describes the sets of information and visualizations that were adapted to the different audiences, involved as decision-makers in the delivery of new Content and Metadata products and platform functionalities.

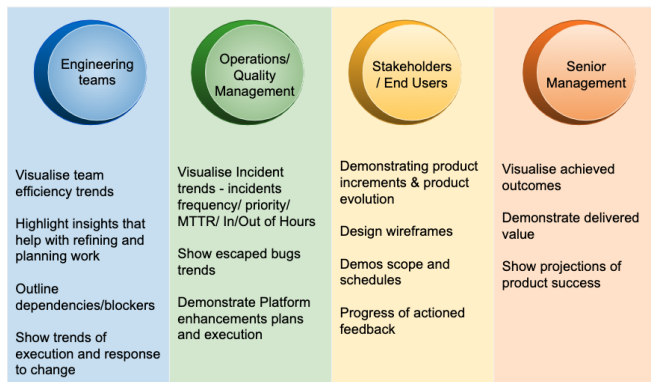


Fig. 2. Delivery analytics visualization adapted to audience specifics

All the above-described delivery analytics were being run in parallel to provide a holistic view of the product development process. The metrics were predictive of product timely completion, quality acceptance, platform uptime, team performance improvement.

The visualized analytics were provided in direct communication with the intended audience in line with the Agile Manifesto value: Interactions with people over processes and tools. The direct communication ensured quick feedback loops. It led to prompt actioning of the decisions advised by the various audiences. The result was an efficient change of direction when needed or staying the course.

Sustaining such a highly collaborative effort proved possible only through a regular cadence of updates. The frequency was tailored per the audience requirements varying from bi-weekly to quarterly updates. It was aligned with the working processes of the different professionals. More strategic roles required a high-level overview, which required data to be gathered over a more extensive period.

Engineering teams needed the most frequent updates. The provided analytics were aligned with the start date of each iteration. Findings and decisions on improvements could be applied right in the next iteration. The results of these improvement decisions could be reviewed as soon as the next iteration ended. This allowed for the freedom to try out bold ideas, new engineering practices, changes in team structures or team processes. This level of transparency created a mindset of continuous improvement. It also influenced team members' intrinsic motivation to develop new ways of efficient development and collaboration on a team and cross-team level.

## V. THE BUSINESS CASE

A more in-depth overview of the proposed analytics-driven agile software product delivery approach is presented in the sections below, with the help of an illustrative business case.

### A. Guiding technical teams

After product and architectural requirements are clarified for the initial stage of development, the technical teams start to break down the technical solution into executable work items. High-level dependencies are highlighted and actioned. Additional research and analysis tasks are logged. This allows for the sequencing of development activities with a span of two iterations. The rest of the work is kept planned at a high-level so that necessary changes can be accommodated promptly.

An important insight at this point is how much work must be planned in one iteration so that the engineering team can complete it at a sustainable pace. Fluctuations in delivery pace are to be expected. However, they would not drastically impact delivery dates of the product increments.

One of the metrics that provide such insight is the team velocity. It is measured in Story Points - a unit of measure that has a specific value and meaning to the team that has assigned it. The velocity trends in Figure 4. are based on story points that represent work/research task complexity only. The Story Points values have been agreed by the whole team during the Planning sessions and have been baselined against the analogous estimation of previously completed work items.

Velocity is impacted by team forming processes for the first few iterations. After reaching the team's performing stage, team capacity is the factor that affects velocity to a great extent. Fig. 3 represents the range of fluctuation of a team of six engineers over seven months.

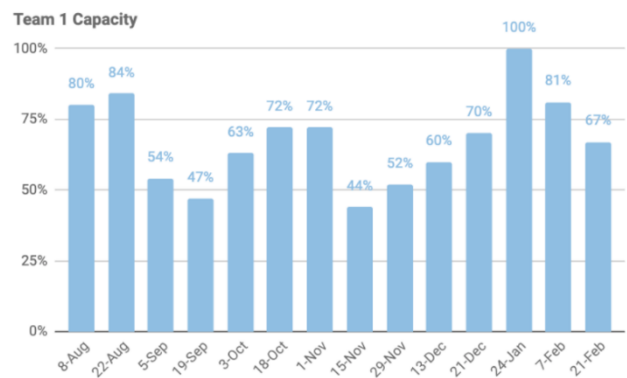


Fig. 3. Capacity Fluctuations over 2-week iterations

Team 1 was trusted with developing a fundamental platform capability – functionality that solved a seven-year-long problem. The value Team 1 delivered had a direct impact on the retention and acquisition of B2B customers. Given the strategic nature of the project, predicting delivery dates based on the assumption of steady capacity would have been detrimental to the business processes, dependent on the project completion.

The visualization in Fig. 4 is showing insightful trends in the planning and execution processes of Team 2.

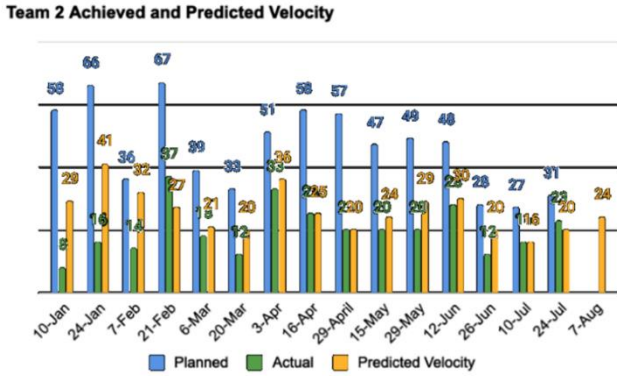


Fig. 4. Velocity predictability over 2-week iterations

Team 2 has over planned upon forming, then managed to increase efficiency and has started to plan less. After increasing the delivery pace, Team 2 has kept its ambitious nature and has started to plan more per iteration. The planning and estimation processes of Team 2 have been revisited three times for the time represented in the chart.

Team capacity has been considered for the value of the Predicted Velocity. Average Velocity has proven to be a more precise metric in comparison to Median Velocity.

Equations (1) and (2) were used to form the values represented in the above chart (Fig. 4).

$$AvV = \frac{1}{n} \sum_{i=1}^n \frac{AcV_i}{TC_i} \quad (1)$$

$$PV = AvV \times TC_{n+1} \quad (2)$$

Where:  
*n* – number of past iterations  
*i* = 1, 2, ..., *n*  
*AvV* – Average Velocity  
*AcV* – Actual Velocity  
*TC* – Team Capacity  
*PV* – Predicted Velocity

The complementary chart on Fig. 5 was used to show if there was any planned work that was not actioned during the iteration.

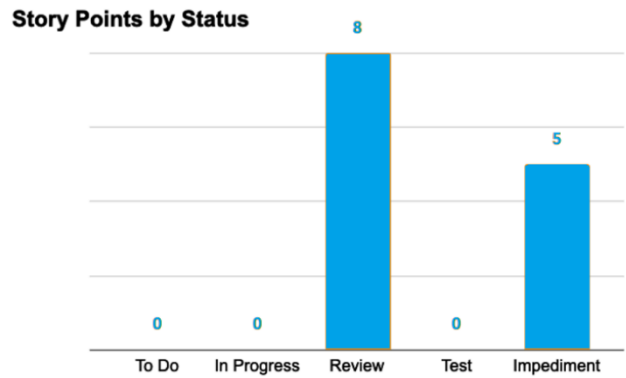


Fig. 5. End of iteration work distribution per status

If any items remained in status ‘To Do’, Team 2 considered reducing the amount of work assignments for the next period to address the spill of work between iterations.

The third chart that drove decisions is shown in Fig. 6. It is a standard Cumulative Flow Diagram (CFD) that shows the amount of story points that are blocked per iteration, due to dependencies on other teams. Team 2 strived to resolve those dependencies. In some iterations the focus of the team was to reduce the amount of work on hold and improve the flow of delivery.

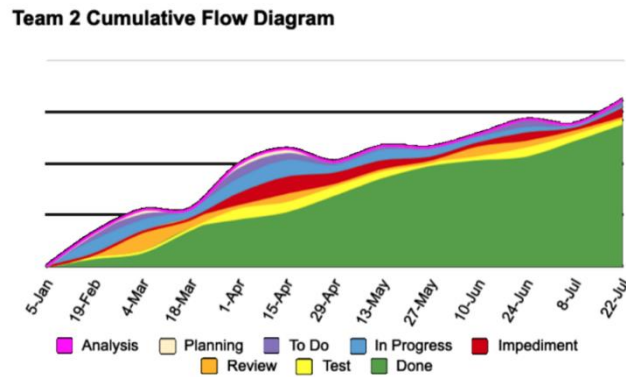


Fig. 6. CFD highlighting work on hold

The following specifics represent the context in which Team 2 was performing:

- Several products had to be developed in parallel
- The product development was in collaboration with other remotely located teams
- Requirements changed in the late stages of the development
- There was no dedicated Product Manager to Team 2, and the Business Analyst executed the Product Owner role
- Releasing product increments to Production required complex integrations between platforms
- There was no dedicated Quality Assurance role
- There were tight deadlines for each product
- Each of the products was of strategic importance to the company, executing on Horizon 2 Product vision (nurturing emerging business)

The results achieved by Team 2 and the collaborating teams were recognized on a company-wide level. Deadlines were met with precision. Impressive quality was achieved: no escaped bugs for any of the delivered products, no incidents

related to the integrations were experienced, security reliability score was maintained at the required level.

### B. Guiding platform teams

The organization, in which this business case study was conducted, has established DevOps for years. In this setup, developing and platform teams (the teams that operate the platforms) work closely together from the initiation of a new product build to ensure security, reliability and feasibility of the systems that will run the new products.[6] The close cooperation with platform operations is the building block that enables revolutionary innovation and premium quality.

Many of the new technical solutions require new platform capabilities that are provisioned by the platform teams. Alignment between development and platform teams can also be guided through delivery analytics.

The insights that are needed for improving the efficiency of the platform teams include system stability and flow of work.

The visualizations that provided those insights are shown in Fig. 7, Fig. 8, and Fig. 9 below.

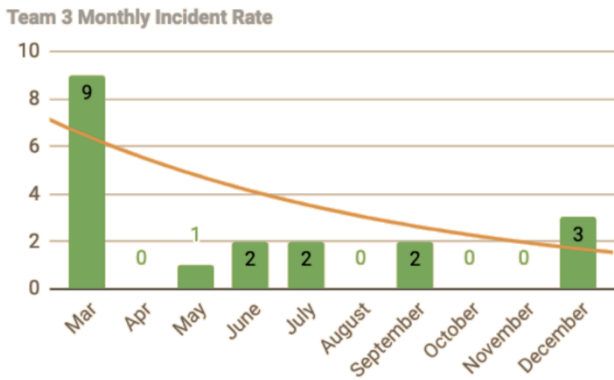


Fig. 7. The monthly incident rate of 2019

The monthly incident chart for 2019 (Fig. 7) provides an overview of the system stability of Team 3 from the first month the team took over responsibility for a complex platform with more than 150 microservices. The capabilities transfer continued for seven months over different time zones and remote training sessions. Development of new platform capabilities and decommission of old functionalities was run in parallel with the knowledge transfer.

The incident rate chart for 2020 shown in Fig. 8 shows the evolution of the inherited platform in terms of stability, reliability, and uptime. This is a direct result of the improved collaboration between the platform and development teams. The number of new products and platform capabilities released in 2020 is almost double as compared to 2019. Yet the platform is proving more stable and ready to host new releases and changes.

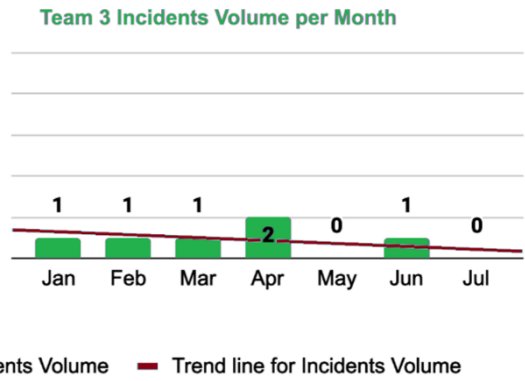


Fig. 8. The monthly incident rate of 2020

Platform teams face the challenge of providing platform capabilities and upgrades in alignment with new product development roadmaps. Ensuring uninterrupted flow of work is crucial for efficient execution of needed system changes. Minimizing dependencies between system components, build of Continuous Integration / Continuous Delivery pipelines, powering continuous deployment capability, provisioning release on-demand functionality are all very complex and voluminous initiatives. Uninterrupted flow and sufficient granularity of work items are vital to executing with time and cost-efficiency.

The chart in Fig. 9 shows the reduction of delivery time for key platform changes performed by the platform team, called Team 4 in this study.

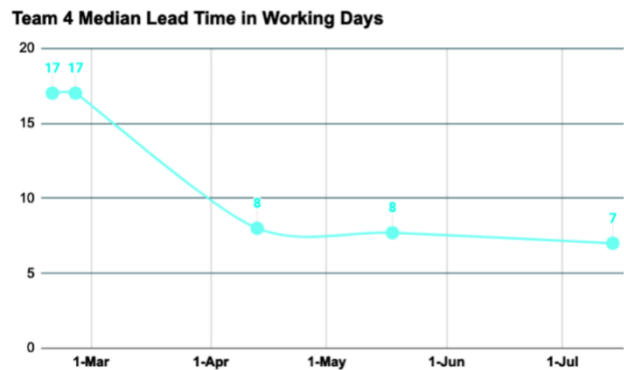


Fig. 9. Delivery time reduction for platform initiatives

The changes that were applied since the beginning of the monitored period include increased granularity of work tasks, adoption of sub-tasks, closer collaboration between team members, reduction of initiatives that run in parallel. The result is a double decrease in time to deliver needed platform enhancements. No product release timelines were missed due to system incompatibility. Security and reliability targets were exceeded significantly every quarter.

### C. Reporting to senior management

Senior Management endorsement of product initiatives needs to be informed by delivery analytics. Sponsoring of strategic products is sometimes done by C-level management directly. This requires information to be provided in a concise and if possible, interactive way.

The provided charts need to also inform the preparation of marketing campaigns that are time and market sensitive.

Fig. 10 and Fig. 11 represent the Roadmap overview execution of Team 2. They were presented through an interactive chart in a reporting tool that was being evaluated at the time of the study. Hovering over each deliverable showed the name of the delivered product feature, increment or version. (Fig. 10)

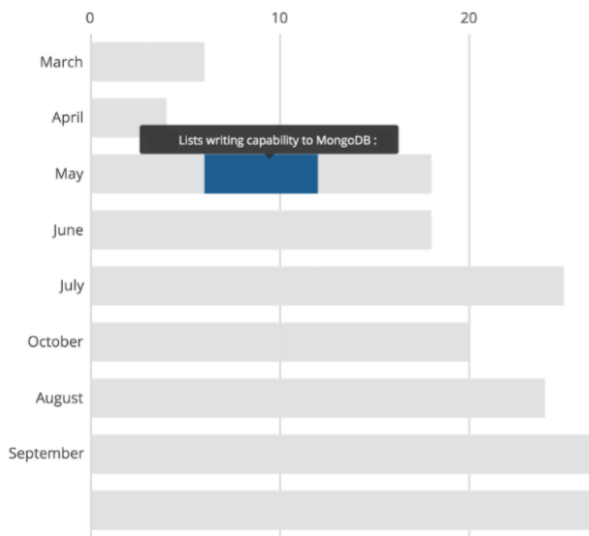


Fig. 10. Team 2 Interactive Roadmap execution overview

An example of a detailed overview of the Roadmap is displayed in Fig. 11. The actual product details have been unshared in line with confidentiality guidelines.

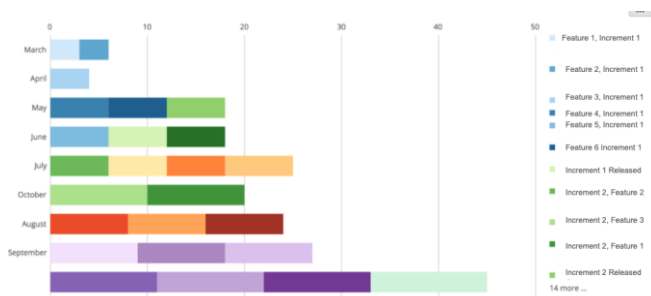


Fig. 11. Team 2 Product Roadmap

These visualizations provided a one-glance view that served the interests of Senior Managers from different business units. The information proved useful for driving internal and external product announcements in line with market cycles and events.

Product funding relies on information about the outcomes achieved through the development of different product increments and versions. This requires translating the value that was gained through the release of new customer offerings to units of value that matter to Senior and C-level management: client engagement, newly registered customers, client lifetime value, customer satisfaction rating, ROI (return on investment), etc.

Meeting targets for these types of tangible value is an important goal for Senior / C-level Management. The visualization in Fig. 12 was used for the evaluation of budget allocation to a 3rd party team that provided a unique service

needed for the development of in-house platform functionalities.

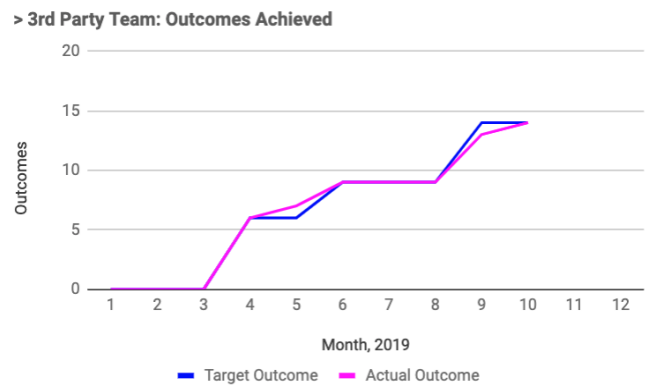


Fig. 12. Value and Outcomes delivered by a 3rd party team to the organization

Based on the quarterly review of the value delivered by the 3rd party team, an additional budget was allocated, so even more innovative products could be built in partnership with that team during the same year.

In all of the above-described cases, the changes that took place as a result of conversations around the visualized delivery analytics were then applied long-term and brought in results of strategic importance.

## VI. RECOMMENDATIONS RESULTED FROM THE STUDY

Crucial to harness the power of delivery analytics is to follow certain principles that would bring on the right behaviors and will epitomize the company's values.

Transparency is one of those fundamental principles. Charts showing the data that feeds into them are better understood and trusted. Switching between data tables and visualization needs to be made easy and swift. This enables seamless adoption of the analytics-driven approach in engineering teams.

Disambiguation of provided analytics is another important principle. The goal of predictive recommendations is to communicate trends that will inform strategic decisions and changes. Direct communication, adapted terminology, the low cognitive load of visualizations are the essential elements that help achieve the needed clarity and effectiveness.

All metrics in scope of the delivery analytics need only to reflect performance and results achieved on team level. Measuring individual contributions create negative behaviors and are not predictive of goals completion.

Following these fundamental principles, aligns well with the values described in the Agile Manifesto. One of the 12 Agile principles states that "Working software is the primary measure of progress". The analytics-driven delivery needs to ensure this remains the guiding light for all engineering teams and stakeholders.

Analytics-powered delivery often leads to changes in work processes. Efficient feedback loops drive changes in the new products' scope. Shifts in scope alter the specifics of development and platform teams. The delivery analytics needs to adapt to the changes it drives. The practical examples of visualizations and resulting changes have proven successful mainly because they were always being adapted to the teams'

and products' needs. Delivery analytics has evolved alongside the evolution of product development.

**Disclaimer—The charts provided in the business case study were used in addition to the company's automated reporting tool.**

## VII. CONCLUSIONS

The presented business case study and the results of the applied methodology lead to the following conclusions:

Agile software product delivery needs to rely on real-time data analytics in the same way as product conceptualization relies on up-to-date market and user data.

Analytics-driven development and platform operations lead to timely delivery of valuable products that run on stable systems and bring in unique value to customers.

Delivery teams' efficiency empowers innovation and boosts quality. Team efficiency is achieved through informed decisions about changes in work processes and engineering practices. The information for those decisions is provided by visualized data sets that are transparent, unambiguous, updated and adapted to the latest team and product specifics.

Team motivation and dedication is impacted by the values represented in the delivery analytics process. The Agile Manifesto and company's values provide practical guidance in that respect.

The goal of an organization is to bring value to all stakeholders: customers, management, employees. The insights from a carefully curated analytics-driven delivery

unlock the potential for increasing that value to unprecedented heights.

Applying analytics-driven agile software product development elevates creativity, inspires new products, and empowers organizations and customers to thrive.

## REFERENCES

- [1] H. Singh, Project Management Analytics, Old Tappan: Pearson Education, 2016.
- [2] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, D. Thomas, 'Manifesto for Agile Software Development,' 2001. [Online]. Available: <https://agilemanifesto.org/> [Accessed: 28 August 2020].
- [3] Agile Alliance, 'Agile Alliance | Glossary | Incremental Development' 2020. [Online]. Available: [https://www.agilealliance.org/glossary/incremental-development/#q=~\(infinite~false~filters~\(postType~\( '~page~'post~'aa\\_book~'aa\\_event\\_session~'aa\\_experience\\_report~'aa\\_glossary~'aa\\_research\\_paper~'aa\\_video\)~tags~\( '~incremental\\*20development\)\)~searchTerm~'~sort~false~sortDirection~'asc~page~1\)](https://www.agilealliance.org/glossary/incremental-development/#q=~(infinite~false~filters~(postType~( '~page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~( '~incremental*20development))~searchTerm~'~sort~false~sortDirection~'asc~page~1)) [Accessed: 28 August 2020]
- [4] Agile Alliance, 'Agile Alliance | Glossary | Iterative Development' 2020. [Online]. Available: [https://www.agilealliance.org/glossary/iterative-development/#q=~\(infinite~false~filters~\(postType~\( '~page~'post~'aa\\_book~'aa\\_event\\_session~'aa\\_experience\\_report~'aa\\_glossary~'aa\\_research\\_paper~'aa\\_video\)~tags~\( '~iterative\\*20development\)\)~searchTerm~'~sort~false~sortDirection~'asc~page~1\)](https://www.agilealliance.org/glossary/iterative-development/#q=~(infinite~false~filters~(postType~( '~page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~( '~iterative*20development))~searchTerm~'~sort~false~sortDirection~'asc~page~1)) [Accessed: 28 August 2020]
- [5] A. Ivanova, P. Ivanov, 'Data Analytics for DevOps Effectiveness' in Proceedings of the 14th International Conference on Computer Science and Education in Computer Science, Boston, MA, USA, 2018
- [6] G. Kim, J. Humble, P. Debois and J. Willis, The DevOps Handbook, Portland: IT Revolution Press, 2016