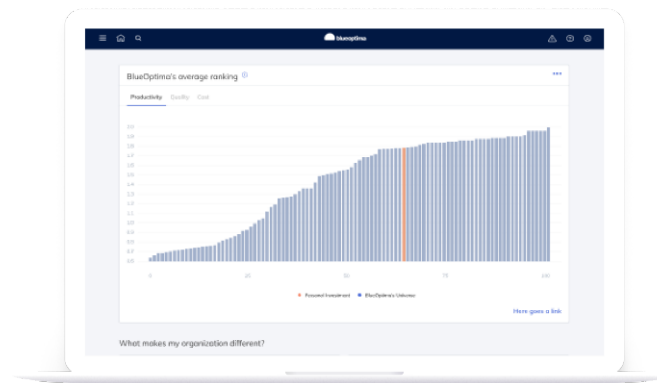# Developer Analytics Case Study

How one of Europe's largest Universal Banks turned a challenging Agile transformation into a successful one.

Introduction

Software development has become a core component for many industries and choosing digital agility has become increasingly important for serving customers in the fastest way. Agile is the ability to create and respond to change rapidly.

Agile software development consists of various iterative and incremental approaches to software development. Under which requirements and solutions evolve through the collaborative effort of self-organising, cross-functional teams with their end-users.

A discipline of Agile continuous improvement. A plan for improving Agile adoption while increasing software development productivity is achieved by measuring the health of an organisation's Agile transformation.

One of Europe's largest Universal Banks soon discovered this after it began an Agile adoption to improve the delivery of digital services when customer demands grew for more online transactions.

A three-year strategic plan was put into place for the IT department with a focus on flexibility and stability with every product deployment.

The IT department adopted Agile principles with DevOps (Development and Operations) best practices. The aim was to improve

engineering teams' ability to respond to customer needs and accelerate time to market.

To increase efficiency, the CTO and Technical Leads set the following objectives:

- Improve capabilities to enhance software quality
- Create a better software development culture
- Enhance delivery predictability

A Case for Agile

Two years into their Agile transformation, it was becoming a challenge to enable continuous deployment to meet customer needs. Software development features were not being delivered on time, under budget or with the expected quality.

To succeed in its Agile mission, the Universal Bank needed to discover a way to measure the health of its Agile transformation, to improve Agile adoption and increase software development productivity. The CTO needed help to understand what practices could be slowing their transformation and the reasons why.

The Universal Bank's IT department used iteration burndown charts and individual hours per iteration per week to measure the success of individual Agile projects. Despite this, they were struggling to improve their software delivery.

The CTO had realised they could not scale their existing measures and lacked actionable insights on what activities need to be refined to ensure a successful transformation. Also, the Universal Bank needed a

plan for tracking and improving Agile adoption while increasing software development productivity.

The CTO had remarked when reflecting on the Agile adoption:

"We aim to follow a customer-centric approach with the goal to better flexibility and stability with every product deployment. Since adopting Agile, our goal has not been met. We were using Agile, but we didn't have team level transparency to make sure our teams were comfortably transitioning from old practices to adopting Agile practices."

BlueOptima's Approach

The CTO became aware of BlueOptima's Developer Analytics and decided to go ahead with a limited rollout at two of their locations in Europe. BlueOptima's Developer Analytics platform provides data- driven insights to improve software development efficiency. At the beginning of the rollout, Developer Analytics was used to provide the CTO with objective insights into how their source code was changing over time, and analyse the impact on engineering teams' productivity.

We measured their current Agile approach to analyse the impact on engineering teams'

Billable Coding Effort Distribution (%)

Avg. BCE/Day: 1.36
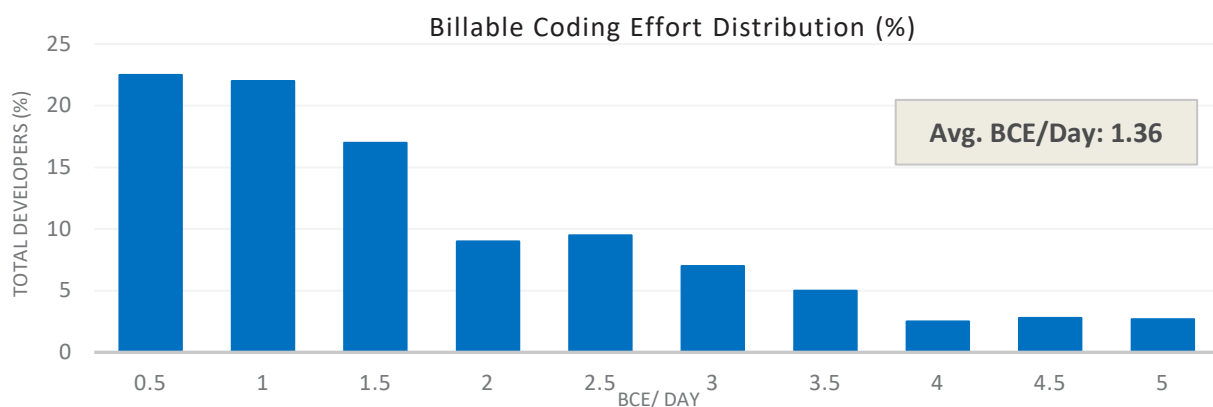
To read the graph: Y-Axis shows the number of developers delivering meaningful source code changes and X-Axis shows productivity distribution buckets with increasing productivity as you move towards the right.

productivity. We were able to give them insight into their teams' adoption of Agile, highlighting which teams were excelling working in an Agile manner, which teams were not successfully transitioning, and which projects were most affected.

This observation gave them a better understanding of current performance levels and plan what performance level they want to achieve. They were able to formulate a plan to restore their Agile transformation. The CTO and Technical Leads were able to set realistic goals for what could be achieved, including the engineering teams in critical need.

A snapshot of the productivity Trends of the Universal Bank's Agile transformation before the

BlueOptima limited rollout began shows that their engineering teams were at 1.36 Avg. BCE/Day

The insights identified showed that the majority of their development resources were underutilised. It showed a small number of developers were delivering the bulk of the source code change within their development workforce.

During the Universal Bank's rollout of Developer Analytics, the CTO and Technical Leads were able to identify which Agile processes were incorrectly implemented among their teams that were negatively impacting the transformation. They were able to adjust their approach and improve the productivity of their engineering teams.

BlueOptima helped them notice a big shift in productivity distribution. They were able to see an increased number of developers moving to higher BCE/Day.

With a 35% increase in the BCE/Day, the Bank's engineering teams were now at 1.83 BCE/Day

The Universal Bank wanted to further explore agile practices that will benefit their engineering teams. The CTO decided to opt for BlueOptima's Data Science package. BlueOptima's Data Science team collaborated with the Universal Bank to create custom reports to indicate the Universal Bank's performance against Agile best-practices.

Here are some of the insights revealed by the custom reports.

Sprint Overview on a Team Level

A key metric in determining the success of an Agile transformation is sprint duration. BlueOptima's Data Science team provided quick, actionable insights with a snapshot of completed sprint activities by engineering teams. Technical Leads analysed sprint-on-sprint productivity to understand the reasons behind the impact on their engineering teams productivity. Therefore, the bank improved and optimised future sprints.
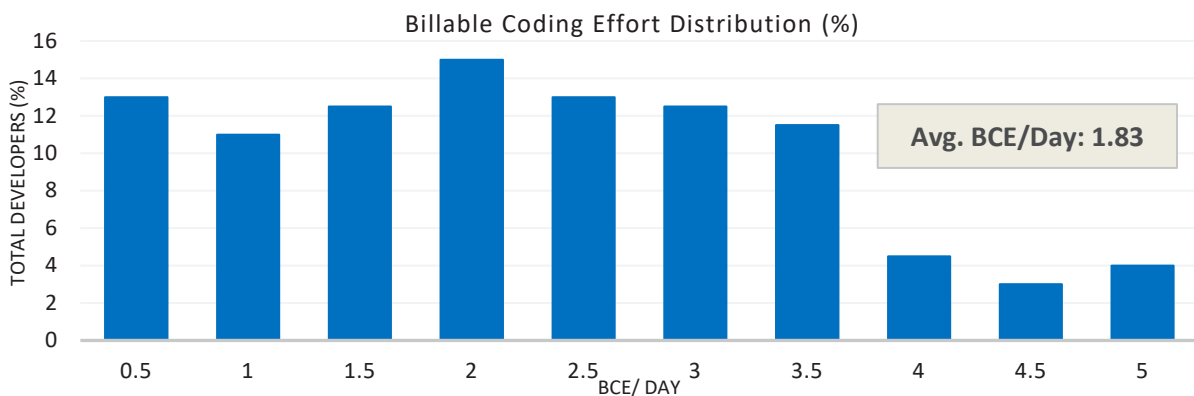
The graphs are a snapshot in time that demonstrates what percentage of developers deliver a range of code, i.e. how many developers delivered less than 1 BCE/Day to 1.5 BCE/Day.

Sprint review
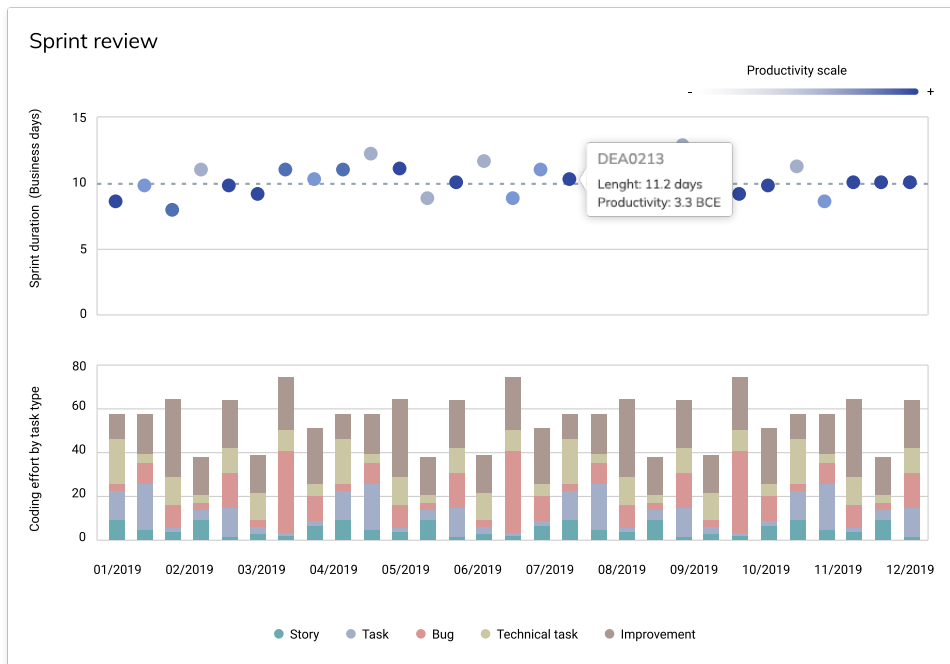
Sprint Overview on an Individual Level

A comprehensive view of all major tasks engineering teams completed during a sprint was provided. Technical leads were able to identify if context switching between tasks was impacting individual developer's productivity. The shapes in the chart represent different task types undertaken by individual developers. The Coding Effort invested in delivering those tasks is reflected in the size of the shapes.

Conclusion

An Agile transformation for this major Universal Bank was a worthwhile investment. They saw noticeable improvements in their engineering teams' productivity with an increased number of developers moving to higher BCE/Day. The engineering teams were now at 1.83 BCE/Day. They achieved a 35% increase in BCE/Day.

The Universal Bank gained a $10,000 per developer in cost reductions due to increased productivity. In this case the 35% increase in BCE/Day the organisation achieved would be equivalent to
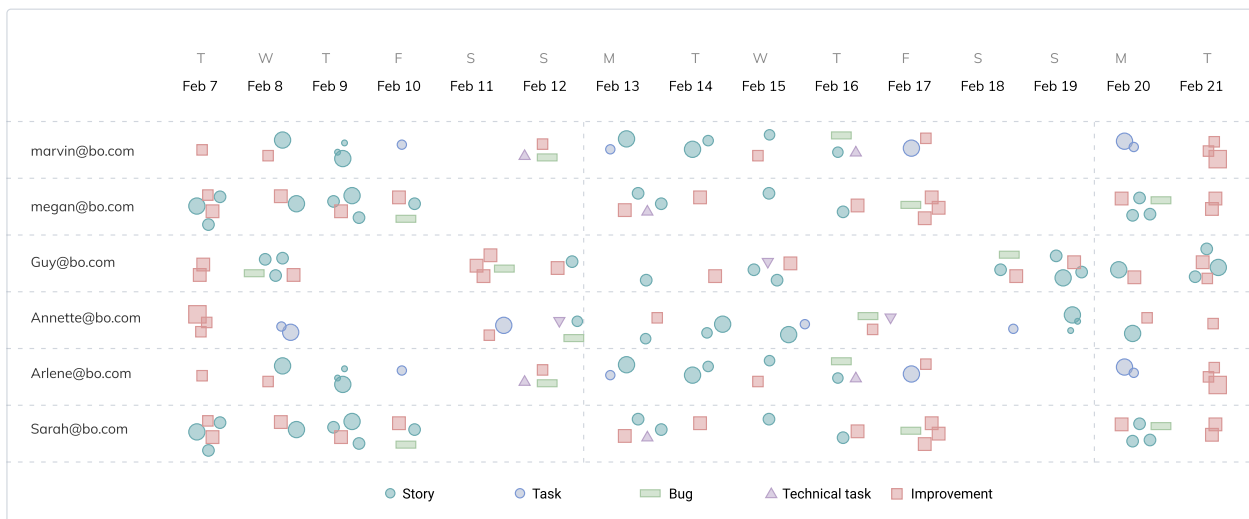
The information presented in this study uses data from standard reports that are created for illustration purposes only to protect confidential client data.

35%

Increase in BCE/Day

$10k

per developer in cost reduction

the work produced by an additional 10% of high performing software developers. They saved cost on hiring or outsourcing by maximising the efficiency of their current software developers.

BlueOptima's solution increased the CTO and Technical Leads access to objective data that helped to increase the teams' compliance with Agile development methods. They were able to better improve their Agile transformation and customer user experience by achieving timely deliveries.

Background on our metrics

BlueOptima analyses the intellectual effort software developers invest when delivering source code changes. Historical revisions are analysed to produce an objective benchmark of the work delivered by software developers called Coding Effort.

This analysis of Coding Effort is done in two parts: Actual Coding Effort (ACE) which is the total effort put into the commit and Billable Coding Effort (BCE) which is the prorated version to represent how that work was done over time.

Actual Coding Effort (ACE)

ACE is an objective, internally benchmarked measure of productivity which calculates the

intellectual effort invested by developers to affect source code change. The analysis is done by evaluating modifications made to 36 source code metrics, broadly grouped across three dimensions: volume, complexity and interrelatedness.

Billable Coding Effort (BCE)

After calculating the Coding Effort, we understand how that translates to Billable Coding Effort. BCE is a proration of time spent coding across working hours, including time spent coding outside of work, on weekends and holidays. We cap BCE at a maximum of 5 hours a day.

Time and motion studies, done in an academic and industry setting, show that 5 hours is the total amount of Coding Effort a developer can consistently deliver within a working day. It accounts for other work activities in a developer's day that does not involve coding, for example daily standups, backlog grooming sessions and retrospective meetings.

Learn more about how BlueOptima can help you deliver Better, Leaner, Safer and Faster

Email Data Science

p +44 207 100 8740

e enquiries@blueoptima.com

w www.blueoptima.com