It takes a Flywheel to Fly: Kickstarting and Growing the A/B testing Momentum at Scale

Aleksander Fabijan
Experimentation Platform (ExP)
Microsoft
Bellevue, WA, USA
Aleksander.Fabijan@microsoft.com

Benjamin Arai
Experimentation Platform (ExP)
Microsoft
Bellevue, WA, USA
bearai@microsoft.com

Pavel Dmitriev
Data Science
Outreach
Seattle, WA, USA
Pavel.Dmitriev@outreach.io

Lukas Vermeer
Experimentation
Booking.com
Amsterdam, Netherlands
Lukas.Vermeer@booking.com

***Abstract*— Companies run A/B tests to accelerate innovation and make informed data-driven decisions. At Microsoft alone, over twenty thousand A/B tests are ran each year helping decide which features maximize user value. Not all teams and companies succeed in establishing and growing their A/B testing programs. In this paper, we explore multiple-case studies at Microsoft, Outreach, Booking.com, and empirical data collected, and share our learnings for iteratively adopting and growing A/B testing. The main contribution of this paper is the A/B Testing Flywheel. This conceptual model illustrates iteratively navigating the value-investment cycle with the goal to scale A/B testing. In every turn of the flywheel, teams need to invest in order to increase the A/B testing momentum. We describe the investments in software development processes that have been advantageous in getting the flywheel to turn. We also share example metrics that track the progress towards sustainable A/B testing momentum.**

Keywords — A/B Testing, Online Controlled Experiments, Flywheel, Data-Driven Culture

I. Introduction

A/B tests are ran to learn how specific changes and features affect users' experience, satisfaction, system performance etc. [1], [2]. Some teams, products and companies run thousands of A/B tests every year. Many, however, rarely experience the full benefit of A/B testing [3]. For example, even within Microsoft we have seen various levels of velocity in adopting and scaling A/B testing. Some product teams that adopted A/B testing quickly made it a standard step in the product development process. For these teams, A/B testing is a business-critical tool for decision making and risk mitigation [4]. For others teams, such growth does not happen or occurs at a slower rate [3], [5], [6]. Why is growing A/B testing so challenging given the advancements in infrastructure [7], [8], statistics [9] and tooling in the last decade?

One reason for this is that expanding the use and capabilities of A/B testing and transforming into a data-driven organization at scale requires up-front investments both in terms of technology as well as human capital. For example, investing in an A/B testing platform and hiring or educating practitioners on how to use it in a trustworthy manner. To support these continued and significant investments it is often necessary to provide a steady demonstration of the value. Like with other novel approaches, to obtain investment we need to show value, but to show value we need investment. The solution that helps with this chicken-and-egg problem is to make the transformation iterative, investing in each of the critical steps incrementally. One way to model this iterative process is through a flywheel. We share a simple flywheel illustrating the iterative nature of investments and demonstrating value on Figure 1.

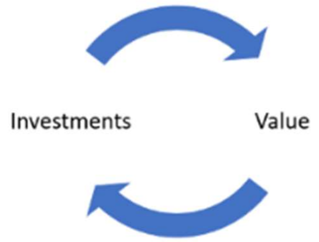

Figure 1. A Simple Flywheel. An initial investment shows value, which drives more investment and more value.

In this paper we build on this idea and introduce the A/B Testing Flywheel – a model for implementing a successful A/B testing program, focusing on iteratively navigating the value-investment cycle with the goal of scaling A/B testing. For the purpose of this discussion, we define a "scaled organization" as an organization that has reached the - "Fly" stage on the Experimentation Maturity model [10], [11] by fully embracing A/B testing for making product decisions. In such organizations, most features deployed to users of the product are evaluated through a valid A/B test.

Flywheels are a proven tool for transforming from good to great [12]. The hardest part about spinning any wheel is getting initial momentum. Many turns may be necessary before momentum carries the wheel with minimal external effort. The interdependence between the steps in a flywheel encourages an organization to continuously improve every step. Neglecting any one step results in friction which can significantly hinder the overall momentum and organizational transformation. For example, if value fails to materialize, investments may decrease. In turn, this leads to fewer valuable A/B tests.

We derived the A/B Testing Flywheel based on our collective experience from introducing and integrating A/B testing to over two dozen product teams at Microsoft, Outreach and Booking.com, and refined it with insights extracted through surveys, tutorials, and other types of collaboration with teams in companies that tried, failed, and succeeded in scaling their A/B testing culture. Our flywheel is a tool for leaders in companies that aim to introduce A/B testing or wish to accelerate its adoption to get value from their A/B testing programs. We enrich the discussion of the flywheel steps with illustrative examples and the cultural and process changes that they helped create at our case companies.

II. Background

A. A/B tests

A/B tests show the causal impact of a change on a product [1], [13]. In the simplest A/B test, users are assigned to use either of the two versions of the product – the control (e.g. the current version of the product) or the treatment (e.g. the current version with an added feature). While the product is being used, telemetry data are collected and later used in statistical tests that will reveal how the customer experience with the two versions differs. Unexpected outcomes are commonly revealed, and we are frequently humbled by our inability to correctly predict the outcome of an A/B test [2]. A/B testing differs from other experimentation techniques such as canary flighting [14] by mandating a control (a version of the product without the new changes).

B. Scaling A/B testing

Previous research inductively derived the Experimentation Growth Model (EGM) from analyzing the experience of growing A/B testing in over a dozen Microsoft products, further detailed through case studies at Skyscanner, Booking, and Intuit [11], [15]. EGM depicts experimentation growth as four stages of evolution, starting from *Crawl* where experimentation is ad-hoc, time-consuming, and provides limited value (e.g. due to the immaturity of the metrics, manual work needed to run and analyze A/B tests, etc.), to *Walk* to *Run* to *Fly*, where experimentation is integral part of every aspect of product development, enabling data-driven decisions at scale. The evolution of experimentation from one stage to the next advances along the seven most critical dimensions: *Technical focus, experimentation platform capability, experimentation pervasiveness, feature team self-sufficiency, experimentation team organization, overall evaluation criteria,* and *experimentation impact*. For a detailed description of these dimensions see [11], [15].

The evaluation of over 60 companies [3] on the EGM as well as the joint research with other companies running A/B testing programs [5] revealed that the biggest challenges in improving along the various dimensions towards the Fly stage were process and culture. Furthermore, for teams or companies that succeeded in scaling A/B testing, sustaining the momentum can be challenging. We illustrate this with an example quote from our survey on experimentation growth:

"We ***no longer*** *have an experimentation culture and* ***not all senior execs are fully bought in*** *to the process."*
– Anonymous survey respondent from www.exp-growth.com.

C. Culture and Processes in A/B testing

Culture is the tacit social order of an organization: It shapes attitudes and behaviors in wide-ranging and durable ways. Cultural norms define what is encouraged, discouraged, accepted, or rejected within a group [16]. A/B testing, when introduced, may challenge several cultural norms such as trusting leaders' experience and intuition over data, people promoting their ideas rather than being skeptical of them, etc. A cultural change will involve a transformation of an organization through multiple phases [17].

A culture of working together towards the common goal of improving products via A/B testing amplifies the benefits of A/B testing at scale. Different approaches for growing A/B testing culture and processes have been recorded in the literature. LinkedIn [5] fosters the culture of experimentation through close guidance. E.g., the LinkedIn experimentation team handpicks a few business-critical teams, prioritizes these teams, and then works closely with them. Over several years a data-driven culture and processes of A/B testing are built across the teams that progressed through this journey. At Netflix a process of peer review of A/B test results is organized around frequent "Product Strategy" forums where results are summarized and debated across the team [5]. Each of these transformations has one thing in common - they were iterative. This brings us to flywheels.

D. Flywheels

The concept of a flywheel was presented by Jim Collins in the book "Good to Great", illustrating that good-to-great transformations never happen at once but in turns [18]. Here is the description of "the flywheel effect":

"In creating a good-to-great transformation, there's no single defining action, no grand program, no single killer innovation, no solitary lucky break, no miracle moment. Rather, it feels like turning a giant, heavy flywheel. Pushing with great effort, you get the flywheel to inch forward. You keep pushing, and with persistent effort, you get the flywheel to complete one entire turn. You don't stop. You keep pushing. The flywheel moves a bit faster. Two turns… then four… then eight… the flywheel builds momentum… sixteen… thirty two… moving faster… a thousand… ten thousand… a hundred thousand. Then at some point – breakthrough! The flywheel flies forward with almost unstoppable momentum."

Flywheels are a proven tool for transforming from good companies to great companies [12]. Each step in a flywheel represents a critical step towards A/B testing maturity. In this paper, we present the A/B Testing Flywheel – a tool for implementing and growing a successful A/B testing program. We derived it based on the method described in the next section.

III. Methodology

This work is primarily a longitudinal case study [19].

Case companies. Microsoft is a large-scale software company with many diverse products across the EGM [11]. At Microsoft, over 20k A/B tests are run every year. Outreach is a startup in the sales domain, providing a sales engagement platform for B2B sales. They have embarked on the journey of A/B testing in 2018 when they ran their first A/B tests. Booking.com is the largest accommodation provider in the world, running tens of thousands of A/B tests every year on their online services.

Data Collection. Our data collection consisted of several qualitative and quantitive data colleciton techniques. The authors of this paper work *as subject matter experts* for scaling A/B testing in their companies and have collected data through action research [20]. Aleksander is a Data Scientist at Microsoft ExP where he works with Benjamin who is a Principal Data Scientist and Program Manager on the ExP team. Pavel is the Vice President of Data Science at Outreach.io, and Lukas is the Director of Experimentation at Booking.com. In aggregate, the authors have over 30 years of experience in the field of A/B testing and data-driven decisions making. Furthermore, we:

- Have been following over 30 product teams that each serve hundreds of millions of users worldwide.
- Employed the EGM for over 3 years. This entailed conducting *surveys* on a continous basis to evaluate how individual teams progress from one stage to the next along the axis. Specifically, practitioners are asked about the blockers, challenges, and solutions for evolving A/B testing. We collected 417 responses in aggregate from practitioners with various levels of expertise in the feld and from different roles (product managers, engineers and data scientists).
- Third, we have been conducting tutorials at conferences (e.g, KDD19, CH2018) where we asked participants to fill-out a public version of the aformentioned survey [21] to evaluate their A/B testing maturity. We collected 258 responses from aprox. 200 companies worldwide.

Data Analysis. The authors of this paper gathered in 15 one-hour online workshops and analyzed the collected data by annotating it and grouping it into categories. The first 4 workshops focused on finding distinct activities that are done to support A/B testing. Then, we codified the activities and categorized them. We started with three predefined categories based on the definition of the EGM (platform, education, value). For example, a training on experimentation platform was annotated as "education". Since an experimentation platform is a prerequisite for this training, this activity was also tagged as an advanced activity. After several iterations, we expanded the themes into five categories which are visible as steps on the A/B Testing Flywheel presented in the next section.

Validity. With respect to construct validity, researchers and participants in this study work in the field of A/B testing and were well aligned on the studied phenomena. Furthermore, half of the 417 participants provided input year over year. In each data collection session, we explained the purpose and terminology at the beginning to all participants. With respect to *external validity*, the results of this paper apply specifically to teams in software companies that are starting to or already run A/B tests and have a goal of scaling A/B testing to the Fly stage of EGM. These would typically be software companies that develop products connected to the internet and have adopted continuous integration and deployment practices [22].

IV. The A/B Testing Flywheel

In this section, we introduce the A/B testing flywheel, discuss how to kickstart it through learnings and examples from our empirical data, and then discuss how to increase its momentum.

A. Introducing the Flywheel

Based on the research described in section III, we derived the following five steps that constitute the A/B Testing Flywheel:

1. **Running more A/B tests to support more decisions.** At the top of the flywheel is the goal – using A/B tests to support decision making. With every turn of the flywheel, we *aim to run more A/B tests to support more decisions*.
2. **Measure value to decision making.** While not every A/B test has substantial impact on decisions, some A/B tests do, and the impact and value added by A/B tests for the customers and business need to be measured and captured. The more A/B tests we run with each turn of the flywheel, the more aggregate value and impact the A/B testing program will provide. If the value of A/B testing is unclear or negative value signals are sent, such as when executives ignore insights from A/B tests, or when employees complain about excessive amounts of time spent configuring and executing an A/B test, it becomes hard to generate interest and make justifications for more resources. Then, the flywheel gets stuck and, without further investment, the A/B testing program will likely remain limited to just a few pockets within the organization. On the other hand, the ability to show strongly and clearly the value of A/B testing sparks interest in other teams and feeds the next step of the flywheel.
3. **Increasing interest in A/B testing.** When A/B testing delivers value for one team, this can lead to increased interest in A/B testing by new teams. To support this spread of interest, dedicated efforts are needed to communicate the value of A/B tests as broadly as possible. Additionally, educational and support efforts are needed to help the individuals and teams who show interest make their initial experience with A/B testing a success.
4. **Investing in A/B testing infrastructure and data quality.** The more interest and willingness there is to try A/B testing within an organization, the more resources can be justifiably allocated to make an A/B testing program successful. These resources should be directed towards two key areas: improving A/B testing platform capabilities for

managing and analyzing A/B tests, and increasing data quality. If we do not have a reliable and trustworthy platform and data, there are a host of issues that can arise - missing data, erroneous statistical calculations, unexpected assignments, etc. - that have the potential to introduce errors, leading to erosion of trust in A/B testing. On the other hand, a well-developed platform enables users to easily execute experiments and understand expected and unexpected results without deep knowledge of A/B testing.

5. **Lowering human (manual) cost of A/B testing.** Improvements in A/B testing infrastructure and data quality create conditions for streamlining the A/B testing process and lowering the cost of manual work i.e., the amount of investment required to start doing A/B testing in a new area. Continuously lowering the cost of A/B testing is a critical step in the flywheel that is often missed. If running A/B tests remains costly, A/B testing will remain limited to highly interested early adopters with a lot of resources. This step increases the return on investment (ROI) of A/B testing by reducing the "I" and making it worthwhile for more teams who may be less convinced about the "R" or are more resource constrained to start running A/B tests.

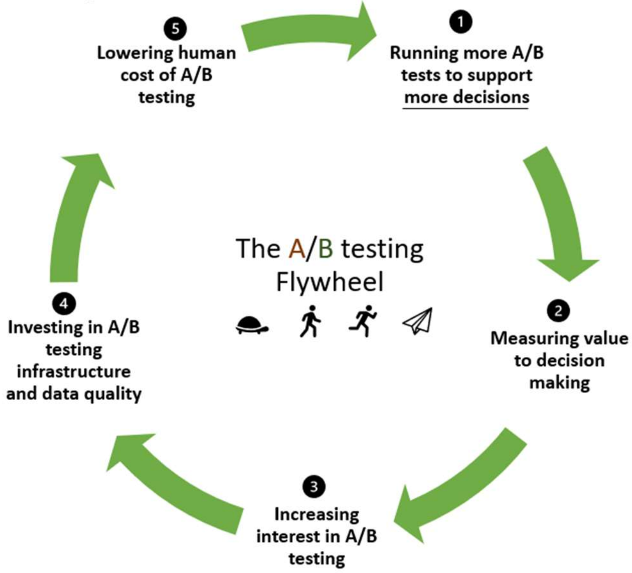
5
Lowering human cost of A/B testing
1
Running more A/B tests to support more decisions
The A/B testing Flywheel
4
Investing in A/B testing infrastructure and data quality
2
Measuring value to decision making
3
Increasing interest in A/B testing

Figure 2. The A/B Testing Flywheel.

Figure 2 depicts the five steps of the flywheel and their relationships. Feeding any part of the flywheel accelerates the loop, speeding up the growth of A/B testing culture in an organization. Conversely, lack of investment in any one of these areas will slow down or stop the growth of A/B testing. Thus, our recipe for successfully growing the culture of A/B testing is simple: push the flywheel – accelerate momentum, then repeat.

In the sections below we discuss each step of the flywheel in more detail – how to get started, what to do to keep increasing the momentum, and how to measure progress.

B. The first turn of the Flywheel

To get the flywheel going, an initial investment will be needed in each of the five steps of the A/B Testing Flywheel.

1. Running the first A/B test. The analysis of our data shows that the growth of A/B testing often starts with a single team trying out A/B testing in one specific scenario, such as the signup flow on a website, or ranking optimization in a shopping app. *In this initial stage, A/B tests need to be chosen wisely.* Initial success may quickly propel the A/B testing program forward, while a failure may stop the program in its tracks before it had a chance to get going. Note that it is important to *differentiate between the success of an A/B test and the success of the idea it was testing*. For example, an A/B test that correctly prevents rolling out a feature that regressed important metrics can be as valuable as an A/B test in which improvements were detected. We found the following factors to be important when selecting initial A/B tests:

- **High value potential.** Focus on A/B testing features that have potential for high impact and relate to team's goals. An A/B test on a small feature in an insignificant area of a website or product is unlikely to deliver significant value.
- **Simple to execute.** Given the lack of A/B testing infrastructure at this stage, initial A/B tests should be as simple to execute as possible. Extra complexities may result in issues with design, execution, and analysis of the A/B test, delaying results and undermining trust in A/B testing. For example, A/B tests requiring coordination across website and mobile app or coordination of several backend and frontend components should be avoided. As a rule of thumb, we recommend a simple A/B test with 1 treatment and 1 control, in a single component/area of the product.
- **Properly powered.** Based on our data and experience, nothing deflates enthusiasm about A/B testing more than the situation when, after all the discussions, predictions about what the results will be, and high expectations for finding out the answer, the results come back inconclusive with no statistically significant changes in the metrics of interest. If there is a doubt about whether the A/B test can be properly powered, choose a different test to run. At this stage, every single A/B test carries a substantial weight, and inconclusive results will pull the flywheel back.
- **Easy to measure.** The success metrics for the A/B test should be easily computable. Ideally, the product should already be instrumented, and data pipelines for metric computation should already exist. It is best to avoid creating new instrumentation and data pipelines *at this stage* and rather validate and reuse existing solutions. Creating new infrastructure requires more time and resources and may introduce data quality issues.

2. Measuring value. Every successful A/B test that demonstrates clear value will push the flywheel forward, while *every failure (e.g. test was underpowered or data could not be trusted) will slow it down or pull it backward*. There is one specific type of value we have seen the most success with for showcasing the power of A/B testing and generating more interest - a *counterintuitive result*. It is a result of an A/B test that contradicts stakeholders' expectations. To obtain a counterintuitive result, pick a test where the outcome is uncertain – there is a disagreement among the stakeholders about what they think the result will be. *By choosing something over which there is a disagreement, we are likely to choose something where test outcome will have value.* If stakeholders disagree about something, it's probably something worth disagreeing about. How to identify such A/B tests? If there are several possible A/B testing ideas on the table, one can do a quick survey of decision makers asking them to predict the result and explain their prediction for each test. One can document the answers and then pick the test with the most disagreement. Once the results are in, someone is bound to be surprised!

For example, consider one of the first A/B tests ran at Outreach. This test added "Just checking in" phrase to the beginning of a follow-up sales email – an email sent a few days after an initial sales outreach email which did not receive a reply. Using the phrase "Just checking in" was considered to be a bad practice in the sales community [23], and not surprisingly the sales leaders and managers who were asked to predict the result all provided predictions in the moderately negative to neutral range. The test result, however, turned out positive – a statistically significant increase in reply rate. This single A/B test helped improve sales content at Outreach, turned sales leaders from skeptics to supporters of A/B testing, and provided valuable PR material to the marketing team – all substantial value adds for multiple different stakeholders [24].

3. Increasing interest. Once an A/B test that provided substantial value is completed, review and share the results as broadly as possible. People we surveyed recommend bringing results to the shiproom, highlighting them at an all-hands meeting, bringing them to the engineering "show and tell", etc. The more venues the better. It is also important to share the results outside of the immediate department. While special means for communicating results of A/B tests may not yet exist in the company in this initial stage, try getting onto the company's newsletter, participating in an internal conference, or giving a knowledge sharing talk to a different group. It is also a great idea to publish results externally. For example, Microsoft's Azure Identity team shared the two variants of sign-up flow they were testing on Twitter, asking people to guess the outcome. They got back many responses, including suggestions on how to further improve the feature [25]. Increased interest may lead to more resources allocated to improving A/B testing infrastructure than the team itself can hope to contribute, making it easier for the team to continue and expand their use of A/B testing.

4. Investing in infrastructure. In the early days of the A/B testing initiative, primary infrastructure focus should be on trustworthiness of A/B tests. Nothing deflates enthusiasm and erodes trust more than an A/B test that had issues that required it to be discarded or reran. One of the most fundamental components for running a trustworthy A/B test is *treatment assignment* [4]. To make sure that it is working correctly A/A tests (A/B tests where both versions are identical) need to be ran regularly [26]. Another key trustworthiness check is a test for Sample Ratio Mismatch (SRM) - an issue that commonly invalidates the results of an A/B test [27]. Separating out treatment assignment, statistical testing, and metric computation into stand-alone reusable components is another key area of focus which ensures that work and improvements that went into one experiment carry over to the next one.

5. Lowering the human cost. Given that trustworthiness of results was mentioned as one of the main challenges at this stage, we recommend to start from automating the fundamental data quality checks mentioned above. Shared libraries for executing trustworthiness checks not only save time, but also help ensure that new problems do not get introduced as infrastructure, data, and metrics evolve over time. A/B testing office hours ran by teams that were successful with A/B testing is another common practice to help those who are just starting out and are employed by all of our case companies.

C. Making the Flywheel Spin

Now that we made the first turn, we need to continue to invest in increasing the flywheel momentum. While the ordering of which step of the flywheel to focus on first and what type of investment to make differed in the collected data, common themes and techniques emerged. We describe them below.

1. Running more A/B tests. Suppose a few initial A/B tests were run successfully. What to do next? Here are common ways to continue the growth:

- **Concurrent A/B tests within the same team/scenario.** Beginning to run A/B tests concurrently is a necessary step of experimentation growth. It requires more A/B testing infrastructure as well as more mature processes for coordinating concurrent A/B tests to avoid interactions.
- **More scenarios within the same team.** For example, if the team started with frontend A/B tests, they could add backend A/B tests. If the team started with using A/B tests to evaluate impact of new features, they can add a scenario of using A/B tests to automatically evaluate feature rollouts [4]. In this way, new applications can be exposed to A/B testing, uncovering more value, and providing more examples for other teams interested in starting A/B testing. As the number of scenarios grows, it is important to avoid technical debt by promptly removing any configurations that are no longer needed [28].
- **Expand existing scenarios to new teams and new product areas.** To keep increasing the flywheel momentum, A/B testing needs to expand from one team to the next. For this to happen, it is very important that a new

team starting A/B testing has a *good role model* – another team which runs "similar" experiments, and which is in a more evolved stage of experimentation growth. While expanding to a new team "similar" to the one already running A/B tests successfully is relatively easy, facilitating an expansion to new product areas and departments, such as from website to mobile, requires improved infrastructure, education, and well-established practice of supporting decisions with A/B tests. These are aspects that other steps of the flywheel need to support. A great way to facilitate the growth of A/B testing in a new area is to transition people with expertise in A/B testing to the new department or area. Our data suggests that this is particularly powerful at the executive level, as it can speed up A/B testing growth by making available resources that normally would only become available at the later stages of A/B testing growth.

- **Broadening the types of decisions impacted**. In addition to evaluating the impact of new features, teams can start running *learning A/B tests*. These are designed specifically with the purpose to learn something about the use of the product, with the goal of informing product decisions or design better metrics. One type of a learning test is to intentionally (but slightly) slow down the load time of the product [29]. By observing changes in business and user metrics in this A/B test, it was possible to assign a precise value to every millisecond of page load time, which both helped inform performance-related product initiatives as well as better inform evaluation of new product features that had load time impact.

The ultimate goal of all these efforts is to create a culture where A/B testing is considered an integral part of making product decisions. A key step in this direction is to start setting team's quarterly and yearly goals in terms of metric improvements, measured via A/B tests. For example, if the team's goal is to achieve a 10% improvement in a certain key metric, then the sum total of all A/B test results the team obtained during that period should add up to over 10%. This approach to goal setting ensures that success metrics are clearly defined and agreed upon in the organization and makes A/B testing the central part of teams' work. In such case A/B testing provides strong influence on the direction of the organization's work.

2. Measuring more value. While the practice of collecting and communicating counterintuitive results should be continued, to fuel continued growth of A/B testing, the focus of value creation needs to shift to *connecting test results to customer and business impact.* Such connection is established by developing a good *Overall Evaluation Criteria* (OEC) – a set of metrics and decision criteria used to judge the outcomes of A/B tests [12]. Defining a good OEC is hard and was reported to be one of the biggest challenges by our survey respondents [3]. A good OEC clearly connected to business goals makes it very hard to ignore the test results and not act on them!

To demonstrate the difficulty of defining a good OEC, consider the challenge of defining an OEC for email templates that one of our case companies – Outreach - needed to solve. Email is the primary channel for sales reps to approach prospective customers, yet simple metrics such as email Open Rate, Click Rate, and Reply Rate are bad indicators of the quality of an email template. For example, a very aggressive email template usually solicits a large number of replies, but those replies are mostly negative or unsubscribe requests [30]. To enable trustworthy evaluation of email template A/B tests and better connect the results to business impact, Outreach developed a machine learning model that classified email replies into several categories such as positive, objection, unsubscribe, etc. Based on this, better metrics such as Positive Reply Rate were defined and made part of the OEC. Another way to increase the value derived from A/B tests is to obtain a more comprehensive understanding of test results by creating a richer set of metrics.

3. Increasing interest even more. With more teams running A/B tests, systematic programs for communicating value and nurturing initial interest need to be created. Below are commonly used approaches observed in our case companies:

- **A/B Testing Newsletter.** Establishing a company-wide newsletter is especially important as a single means for everyone interested in A/B testing to keep in touch, learn about new infrastructure developments, participate in talks, conferences and trainings, learn about interesting experiment results, participate in games and contests to guess results of currently running experiments, etc.
- **Champions program.** An A/B testing champion is a person who has a profound interest in A/B testing as a decision-making tool for teams and leadership. This person acts as the champion for experimentation in their team or department by enabling teams to run A/B tests, communicating teams' needs to the developers of the A/B testing platform, and advocating for experimentation with the leadership. Many participants of our interviews reported presence of a champion within a team as the single most important factor determining success of experimentation growth within that team.
- **Talks and conferences.** To make it easier for teams to share their interesting results and exchange ideas with others, formal venues to support these activities can be created. Microsoft runs "Best Experiments of the Month" series of talks, and organizes bi-annual conference on A/B testing.
- **A/B testing classes.** Introductory courses on A/B testing should be held as often as needed, covering the basic concepts as well as the tooling that experimenters have available. At Microsoft, such intro courses are offered on a monthly cadence and open to all employees. The curriculum covers motivation for experimentation, an overview of key statistical concepts used in an experiment, and an explanation of the features in the experimentation platform that will address common user scenarios. Example historical A/B tests where audience guesses the best outcome are the most popular component of the course as they illustrate not only the diversity of A/B tests that are ran across the

company but also reaffirm the difficulty of predicting the right outcome. For those interested in becoming champions, advanced courses are offered as well, for example a metric design patterns course that teaches participants how to design good OECs.

4. Investing in infrastructure and data quality. Investment in this step of the flywheel is, perhaps, the most important for accelerating flywheel's momentum. The goal is to make it easier and easier to execute a successful A/B test.

- **A/B testing platform.** A common platform that has the capability to manage A/B testing operations across many teams and types of tests is instrumental for the growth of A/B testing. While some companies, such as Microsoft and Booking.com, built their in-house experimentation platforms from scratch, other companies started by using vendor solutions and then built on top of them or over time replaced components of those solutions with the ones built in-house. For example, Outreach built their experimentation solution on top of Launch Darkly [1] . Eventually, all experimentation platforms end up supporting a similar feature set. We extensively published about Microsoft's Experimentation Platform (ExP) and features that are valuable for an A/B testing platform in our papers (see e.g. [8], [31], [32]) and hence will not go into the details here. What is important is that any steps that can be taken to reduce the need for manual intervention allow for different steps of the A/B testing process to be democratized and reduce the overall support cost.
- **Automation of data validation.** Computing A/B testing metrics often requires bringing together several different data sources, such as product usage and revenue, which are constantly changing and evolving. As experimentation platform matures, the main source of A/B testing errors becomes incorrect data. To combat this issue, participants of our study recommended explicitly defining schemas for data streams and automatically validating incoming data against these schemas. Another solution in use at Microsoft is a system that allows users to define a set of validations – invariants that always need to be true for a given data stream – and runs them regularly against the data streams, alerting if any of the tests do not pass.
- **Simplifying and automating metric management.** As more and more teams start running more and more experiments, they need to continuously create, update, and compute more and more metrics across more and more datasets. If metric computation logic is hardcoded in data pipeline scripts, then adding and updating these metrics takes significant time and effort and is very error prone. For example, Microsoft Bing's experiment scorecards consist of several thousand metrics, with no single person understanding the logic behind all these metrics. At Microsoft, this problem was solved with a specially-defined metrics definition language that allows to manage metrics as configurations rather than code, and an accompanying system that compiles metric definitions into experiment scorecard scripts for several popular large scale computation platforms in use at Microsoft [33]. This platform allows anyone, including those with no software engineering skills, to understand, add, and update A/B testing metrics.

5. Lowering the human cost of A/B testing even more. The infrastructure and data quality improvements mentioned above contribute a great deal to lowering the cost of A/B testing by saving time and eliminating errors. Below we list several other initiatives we extracted from our data.

- **Making A/B testing results intuitive** to practitioners who might not be statisticians. This includes efforts to help avoid misinterpretation of results and prevent overlooking of important details. Color coding and summarization of results are two approaches that help with the interpretation [34]. Investing in ways to guide experiment analysis step-by-step through summarization of experiment results will drive down support costs and increase the trustworthiness of decisions making. For example, at Outreach A/B tests are ran by salespeople, and hence great efforts are taken to simplify each step of the process, including automated test evaluation and decision recommendation.
- **Standardized process** for supporting the A/B testing lifecycle. A human process should be used to complement infrastructure improvements. Checklists can be useful for assuring consistency and quality of both the design as well as the analysis of experiments [33]. One of the most common implementations of a checklist at Microsoft is an Entrance and an Exit review. An entrance review is used to organize experiment information before an A/B test is started. Two key data-points that we recommend capturing besides a description of the change that is being evaluated are its expected impact on key metrics and the target group with which the change will be tested. The information collected during an experiment is then summarized in the Exit review where the outcome of an A/B test is compared to the expectations provided in the entrance review. The value of Entrance and Exit reviews are trifold. First, they increase the quality of A/B tests. Second, they are a database serving as institutional knowledge of ongoing and completed A/B tests. Third, they are a great training ground for new people interested in A/B testing.
- **Champions program.** We discussed above the benefits of champions program for increasing interest in A/B testing in new teams. Champions also contribute to lowering the cost of A/B testing by providing easily accessible A/B testing expertise within the team.

D. Measuring the Flyweel Velocity

How does one know that they are making progress and increasing the momentum? While Number of A/B Tests is a

[1] https://launchdarkly.com/

natural metric that comes to mind, it is important to measure the strength of each step of the flywheel, both to understand impediments to growth and to evaluate effectiveness of different initiatives targeted at specific steps. Below we discuss metrics commonly used to evaluate each step of the flywheel.

1. **Running more A/B tests to support more decisions.** *Number of A/B Tests* ran in a month is a metric that is easy to compute and explain. Most teams that participated in our study use this metric. However, many organizations also developed more nuanced metrics. *Number of Valid A/B Tests* counts only those A/B tests that had trustworthy results, and *Number of Quality A/B Tests* counts only those A/B tests that were trustworthy and impacted key metrics in a significant way. Both these metrics can be broken down by the type of test, such as new feature, safe rollout or learning experiment. *Fraction of Validated Feature Releases* counts the proportion of released features that had an A/B test ran. This metric is harder to compute (e.g. counting code check-ins with and without an A/B test is needed for this), but allows to assess how pervasive A/B testing practice is in an organization. Another way to measure the engagement with A/B testing is to measure the proportion or count of team members engaging with the experimentation platform. This is a measure of how widespread A/B testing is at a team and company level.
2. **Measure value to decision making.** How can one measure improvement in the amount of value A/B testing programs generate? In the early stages, simple measurements such as *Number of High-Value A/B Tests* ran and *Number of Documented Success Stories* collected from individuals and teams that ran A/B tests may be sufficient. As A/B testing programs mature, tracking value manually for each A/B test becomes infeasible, and success metrics need to focus on aggregate business impact of A/B testing. Two examples of metrics that help capture it are *Number of Quality A/B Tests* and *Number of Bugs Identified* via experiments. One aspect of the value provided by A/B testing is preventing one from making a wrong decision. This can be captured by measuring *Number of Times a Feature was Not Shipped* because of A/B testing results. If an organization agreed on an all-up OEC, then the total sum of the *Impact on OEC* from all the A/B tests ran during, say, a quarter is another metric.
3. **Increasing interest in A/B testing.** There are many metrics that can be used to measure the success of this step. One of them is *Number of Shared A/B Tests* – which measures the number of A/B tests highlighted to other teams via one of the provided communication channels such as newsletter. Another metric is the *Number of Training Attendees*, counting people who attended the intro and/or advanced courses. Separately we may want to measure *A/B Testing Literacy* of a team/group/product area, by counting the % of people working in that area who attended at least one training course, or % of people listed as experiment owner or stakeholder on at least one A/B test. If we have a Champions program running we will want to track the *Number of Champions*, especially if we have badges to recognize them making it easy to track.
4. **Investing in A/B testing infrastructure and data quality.** This step has the highest and most diverse number of metrics, as is typical for a complex engineering system used heavily across the organization. Aspects typically measured for this step are usage, scale, reliability, performance, usability, level of automation. Metrics include *Number of Experiment Owners*, *Number of Scorecard Users*, *Number of Experiments* executed, *Number of Alerts* delivered, *Number of Alerts Engaged*, *Time to Resolve an Alert*, *Number of Scorecards* generated, *A/B Test Failure Rate (e.g. underpowered A/B tests or tests with data quality issues)* , Platform *Availability*, *Page Load Time* for different pages, *Time from Experiment Start to Live*, *Time from Experiment End to Scorecard*, *Number of Help Tickets* submitted, *Time to Complete* each step of experiment lifecycle (see step 5), *Number of User Tasks Automated*, and data availability.
5. **Lowering human cost of A/B testing.** Two aspects of human cost are time spent and expertise required to execute different steps of experiment lifecycle [35]. While some measurements of these aspects can be captured by the experimentation platform, e.g. *Scorecard Compute Time* and distribution of A/B test compute jobs, most participants of our study measure them via a periodic survey. This results in a series of metrics such as *Time to Configure an A/B Test*, *Time to Add a New Metric*, *Time to Obtain a Scorecard*, etc. for the time spent aspect, and *Fraction of A/B Tests Completed without Expert Help* for the expertise required.

E. Summary

Our learnings show that cultural and process change needed to introduce and scale an A/B testing program is substantial. And technology, while important, is not sufficient to make it happen. Moreover, the change does not happen overnight. Instead, many small shifts in the product development process are needed over time. The five steps of the A/B Testing Flywheel that we derived are the key areas of investment creating conditions for the A/B testing program to grow. We note that, while each step is vital to attend to in the long run, it is not necessary to invest into all these steps equally and all at once. Similarly, it is not necessary that every single experiment results in improvements in each step. Rather, at any given time, software practitioners need to focus their efforts on the weakest step that needs attention the most.

We summarize the key points from the study in Table 1 next.

Table 1. Key takeaways from the A/B Testing Flywheel

Step	First spin	Increasing momentum	Measuring progress

**1.
Run more A/B tests**
	Choose A/B tests that have high value potential, simple to execute, properly powered, easy to measure	Concurrent A/B tests within a single scenario;
		new teams, new scenarios, new types of decisions;
		quarterly/yearly goals measured via A/B tests
			Number of A/B tests;
			Number of Valid A/B tests;
			Number of Quality A/B tests;
			Fraction of Validated Feature Releases

**2.
Measure Value**
	Look for counterintuitive results	Agree on an OEC;
		define comprehensive sets of Local and Diagnostic metrics;
		tap into new types of value
			Number of High Value A/B tests;
			Number of Quality A/B tests;
			Number of Bugs Identified;
			Impact on OEC

**3.
Increase interest**
	Utilize existing knowledge sharing venues: shiproom, all hands, newsletters, conference; publish results externally	A/B testing newsletter;
		Champions program;
		A/B testing classes
			Number of Shared A/B tests;
			Number of Training Attendees;
			Number of Champions;
			A/B testing Literacy

**4.
Invest in infra & data quality**
	Focus on trustworthiness of results;
	run A/A and SRM tests;
	Develop reusable libraries
		A/B testing platform;
		Automated data validation;
		Simplifying and automating metric management
			Measure usage, scale, reliability, performance, usability, level of automation of A/B testing lifecycle

5. Lower human cost	Automate trustworthiness checks; advertise use of shared libraries; introduce A/B testing office hours	Intuitive presentation/summarization of results; standardized process for supporting experiment lifecycle; champions program	Measure human time spent and expertise required: Time to Configure A/B test, Time to Add New Metric, Time to Obtain Scorecard, Fraction of Experiments Completed without expert help

In our study we saw great diversity in the order and the specific tasks executed to improve different steps of the flywheel, reflecting the differences in cultural and technical challenges different organizations faced. Investing too much into a single step of the flywheel at the expense of other steps can be counterproductive and does not necessarily lead to sustainable growth. One of the most common inhibitors that we observed is reinforcement bias: once initial success is achieved by improving a certain step, there is a tendency to focus all the efforts on this step as opposed to other steps where the progress is slower. However, the speed with which the flywheel turns is determined by the weakest step, not the strongest one.

V. Conclusion

The research presented in this paper – The A/B Testing Flywheel - enabled Microsoft, Outreach and Booking.com to scale A/B testing for many products in various stages of user growth, maturity in data pipelines and infrastructure, and the ability to adopt data-driven decisions. We hope that this iterative nature of the A/B testing Flywheel will be helpful to everyone else that strives to kickstart or grow their A/B testing program. In our future work, we will focus on anti-patterns: the inhibitors that we observed which slow down the A/B testing Flywheel momentum, and strengthening each of the steps.

Acknowledgements

We would like to acknowledge everyone at Microsoft, Booking.com and Outreach.io that contributed to this research. Thank you also to all survey participants and reviewers.

References

[1] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "The Benefits of Controlled Experimentation at Scale," in *Proceedings of the 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2017, pp. 18–26.

[2] R. Kohavi and S. Thomke, "The Surprising Power of Online Experiments," *Harvard Business Review*, no. October, 2017.

[3] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "Online Controlled Experimentation at Scale: An Empirical Survey on the Current State of A/B Testing," in *Proceedings of the 2018 44rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2018.

[4] T. Xia, S. Bhardwaj, P. Dmitriev, and A. Fabijan, "Safe Velocity: A Practical Guide to Software Deployment at Scale using Controlled Rollout," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, 2019, pp. 11–20.

[5] S. Gupta *et al.*, "Top Challenges from the first Practical Online Controlled Experiments Summit," *ACM SIGKDD Explor. Newsl.*, 2019.

[6] E. Lindgren and J. Münch, "Software development as an experiment system: A qualitative survey on the state of the practice," in *Lecture Notes in Business Information Processing*, 2015, vol. 212, pp. 117–128.

[7] D. Tang, A. Agarwal, D. O. Brien, M. Meyer, D. O'Brien, and M.

Meyer, "Overlapping experiment infrastructure," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '10*, 2010, p. 17.

[8] S. Gupta, L. Ulanova, S. Bhardwaj, P. Dmitriev, P. Raff, and A. Fabijan, "The Anatomy of a Large-Scale Experimentation Platform," in *2018 IEEE International Conference on Software Architecture (ICSA)*, 2018, no. May, pp. 1–109.

[9] A. Deng, J. Lu, and J. Litz, "Trustworthy Analysis of Online A/B Tests: Pitfalls, Challenges and Solutions," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, 2017, pp. 641–649.

[10] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "The Evolution of Continuous Experimentation in Software Product Development," in *Proceedings of the 39th International Conference on Software Engineering ICSE'17*, 2017.

[11] A. Fabijan, P. Dmitriev, C. McFarland, L. Vermeer, H. Holmström Olsson, and J. Bosch, "Experimentation growth: Evolving trustworthy A/B testing capabilities in online software companies," *J. Softw. Evol. Process*, p. e2113, Nov. 2018.

[12] J. Collins, "Turning the Flywheel: Why Some Companies Build Momentum… And Others Don't." London: Random House Business, 2019.

[13] R. Kohavi, A. Deng, B. Frasca, T. Walker, Y. Xu, and N. Pohlmann, "Online controlled experiments at large scale," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13*, 2013, p. 1168.

[14] G. Schermann, J. Cito, P. Leitner, U. Zdun, and H. C. Gall, "We're Doing It Live: A Multi-Method Empirical Study on Continuous Experimentation," *Inf. Softw. Technol.*, Mar. 2018.

[15] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "The Evolution of Continuous Experimentation in Software Product Development: From Data to a Data-Driven Organization at Scale," in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, 2017, pp. 770–780.

[16] B. Groysberg, J. Lee, J. Price, and J. Y. J. Cheng, "The leader's guide to corporate culture," *Harvard Business Review*. 2018.

[17] "A/B Testing at Scale Tutorial Strata 2018," 2019. [Online]. Available: https://exp-platform.com/2018StrataABtutorial/.

[18] J. Collins, "Good to great: Why some companies make the leap … and others don't," *Meas. Bus. Excell.*, vol. 7, no. 3, pp. 4–10, 2003.

[19] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empir. Softw. Eng.*, vol. 14, no. 2, pp. 131–164, 2008.

[20] A. B. Sandberg, "Agile Collaborative Collaboration," *IEEE Comput. Soc.*, vol. 28, no. 4, pp. 74–84, 2011.

[21] P. Dmitriev and A. Fabijan, "Experimentation Growth," 2017. [Online]. Available: https://www.exp-growth.com.

[22] J. Bosch, "Building products as innovation experiment systems," *Lect. Notes Bus. Inf. Process.*, vol. 114 LNBIP, pp. 27–39, 2012.

[23] L. Ye, "30 Better Alternatives to the 'Just Checking In' Email." [Online]. Available: https://blog.hubspot.com/sales/just-checking-in-follow-up.

[24] P. Dmitriev, "'Just Checking In' - Does It Actually Work?" [Online]. Available: https://www.outreach.io/blog/just-checking-in-does-it-actually-work.

[25] "Microsoft Identity sign-up flow A/B test." [Online]. Available: https://twitter.com/azuread/status/1255957652976422914.

[26] "p-Values for Your p-Values: Validating Metric Trustworthiness by Simulated A/A Tests." [Online]. Available: https://www.microsoft.com/en-us/research/group/experimentation-platform-exp/articles/p-values-for-your-p-values-validating-metric-trustworthiness-by-simulated-a-a-tests/.

[27] A. Fabijan, "Diagnosing Sample Ratio Mismatch in A/B Testing." [Online]. Available: https://www.microsoft.com/en-us/research/group/experimentation-platform-exp/articles/diagnosing-sample-ratio-mismatch-in-a-b-testing/.

[28] K. Kevic, B. Murphy, L. Williams, and J. Beckmann, "Characterizing Experimentation in Continuous Deployment: A Case Study on Bing," in *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*, 2017, pp. 123–132.

[29] R. Kohavi, D. Tang, and Y. Xu, *Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing*. Cambridge University Press.

[30] "Outreach Sales." [Online]. Available: https://www.outreach.io/blog/sales-bro-vs-sales-pro-sales-messaging.

[31] A. Deng, U. Knoblich, and J. Lu, "Applying the Delta method in metric analytics: A practical guide with novel ideas," no. March, Mar. 2018.

[32] W. Machmouchi, S. Gupta, R. Zhang, and A. Fabijan, "Patterns of Trustworthy Experimentation: Pre-Experiment Stage," 2020. [Online]. Available: https://www.microsoft.com/en-us/research/group/experimentation-platform-exp/articles/patterns-of-trustworthy-experimentation-pre-experiment-stage/.

[33] C. Boucher, U. Knoblich, D. Miller, S. Patotski, A. Saied, and V. Venkateshaiah, "Automated metrics calculation in a dynamic heterogeneous environment." 2019.

[34] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "Effective Online Experiment Analysis at Large Scale," in *Proceedings of the 2018 44rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2018.

[35] A. Fabijan, P. Dmitriev, H. H. Olsson, and J. Bosch, "The Experiment Lifecycle," *Accept. to Appear IEEE Softw.*, 2018.