*** To appear at KDD 2013. Available at http://bit.ly/ExPScale ***

Online Controlled Experiments at Large Scale

Ron Kohavi, Alex Deng, Brian Frasca, Toby Walker, Ya Xu, Nils Pohlmann
Microsoft, One Microsoft Way, Redmond, WA 98052
{ronnyk, alexdeng, brianfra, towalker, yaxu, nilsp}@microsoft.com

ABSTRACT

Web-facing companies, including Amazon, eBay, Etsy, Facebook, Google, Groupon, Intuit, LinkedIn, Microsoft, Netflix, Shop Direct, StumbleUpon, Yahoo, and Zynga use online controlled experiments to guide product development and accelerate innovation. At Microsoft's Bing, the use of controlled experiments has grown exponentially over time, with over 200 concurrent experiments now running on any given day. Running experiments at large scale requires addressing multiple challenges in three areas: cultural/organizational, engineering, and trustworthiness. On the cultural and organizational front, the larger organization needs to learn the reasons for running controlled experiments and the tradeoffs between controlled experiments and other methods of evaluating ideas. We discuss why negative experiments, which degrade the user experience short term, should be run, given the learning value and long-term benefits. On the engineering side, we architected a highly scalable system, able to handle data at massive scale: hundreds of concurrent experiments, each containing millions of users. Classical testing and debugging techniques no longer apply when there are billions of live variants of the site, so alerts are used to identify issues rather than relying on heavy up-front testing. On the trustworthiness front, we have a high occurrence of false positives that we address, and we alert experimenters to statistical interactions between experiments. The Bing Experimentation System is credited with having accelerated innovation and increased annual revenues by hundreds of millions of dollars, by allowing us to find and focus on key ideas evaluated through thousands of controlled experiments. A 1% improvement to revenue equals more than $10M annually in the US, yet many ideas impact key metrics by 1% and are not well estimated a-priori. The system has also identified many negative features that we avoided deploying, despite key stakeholders' early excitement, saving us similar large amounts.

Categories and Subject Descriptors
G.3 Probability and Statistics/Experimental Design: controlled experiments, randomized experiments, A/B testing.

General Terms
Measurement, Design, Experimentation, Big Data

Keywords
Controlled experiments, A/B testing, search, online experiments

1. INTRODUCTION

Many web-facing companies use online controlled experiments to guide product development and prioritize ideas, including Amazon [1], eBay, Etsy [2], Facebook, Google [3], Groupon, Intuit [4], LinkedIn, Microsoft [5], Netflix [6], Shop Direct [7], StumbleUpon [8], Yahoo, and Zynga [9]. Controlled experiments are especially useful in combination with agile development, Steve Blank's Customer Development process [10], and MVPs (Minimum Viable Products) popularized by Eric Ries's Lean Startup [11]. In a "Lean Startup" approach, "businesses rely on validated learning, scientific experimentation, and iterative product releases to shorten product development cycles, measure progress, and gain valuable customer feedback" [12].

Large scale can have multiple dimensions, including the number of users and the number of experiments. We are dealing with Big Data and must scale on both dimensions: each experiment typically exposes several million users to a treatment, and over 200 experiments are running concurrently. While running online controlled experiments requires a sufficient number of users, teams working on products with thousands to tens of thousands of users (our general guidance is at least thousands of active users) are typically looking for larger effects, which are easier to detect than the small effects that large sites worry about. For example, to increase the experiment sensitivity (detectable effect size) by a factor of 10, say from 5% delta to 0.5%, you need $10^2 = 100$ times more users. Controlled experiments thus naturally scale from small startups to the largest of web sites. Our focus in this paper is on scaling the **number** of experiments: how can organizations evaluate more hypotheses, increasing the velocity of validated learnings [11], per time unit.

We share our experiences, how we addressed challenges, and key lessons from having run thousands of online controlled experiments at Bing, part of Microsoft's Online Services Division. Microsoft's different divisions use different development methodologies. Office and Windows follow Sinofsky's long planning and execution cycles [13]. Bing has thousands of developers, program managers, and testers, using online controlled experiments heavily to prioritize ideas and decide which changes to ship to all users. Bing's Experimentation System is one of the largest in the world, and pushes the envelope on multiple axes, including culture, engineering, and trustworthiness. In the US alone, it distributes traffic from about 100 million monthly users executing over 3.2B queries a month [14] to over 200 experiments running concurrently. Almost every user is in some experiment: 90% of users eligible for experimentation (e.g., browser supports cookies) are each rotated into over 15 concurrent experiments, while 10% are put into a holdout group to assess the overall impact of the Experimentation System and to help with alerting.

Analysis of an experiment utilizing 20% of eligible users (10% control, 10% treatment) over 2 weeks processes about 4TB of data to generate a summary scorecard. With about 5 experiments in each one of 15 concurrent experimentation areas (conservative numbers), users end up in one of $5^{15} \approx 30$ billion possible variants of Bing. Automated analyses, or scorecards, are generated on clusters consisting of tens of thousands of machines [15] to help guide product releases, to shorten product development cycles, measure progress, and gain valuable customer feedback. Alerts fire automatically when experiments hurt the user experience, or interact with other experiments. While the overall system has significant costs associated with it, its value far outweighs those

KDD'13, August 11–14, 2013, Chicago, Illinois, USA.

ACM 978-1-4503-2174-7/13/08…$15.00.

costs: ideas that were implemented by small teams, and were not even prioritized high by the team implementing them, have had surprisingly large effects on key metrics. For example, two small changes, which took days to develop, each increased ad revenue by about $100 million annually [16].

1.1 Motivating Example

We begin with a motivating visual example of a controlled experiment that ran at Bing. The team wanted to add a feature allowing advertisers to provide links to the target site. The rationale is that this will improve ads quality by giving users more information about what the advertiser's site provides and allow users to directly navigate to the sub-category matching their intent. Visuals of the existing ads layout (Control) and the new ads layout (Treatment) with site links added are shown in Figure 1 below.

Figure 1: Ads with site link experiment. Treatment (bottom) has site links. The difference might not be obvious at first but it is worth tens of millions of dollars

In a controlled experiment, users are randomly split between the variants (e.g., the two different ads layouts) in a persistent manner (a user receives the same experience in multiple visits). Their interactions with the site are instrumented and key metrics computed. In this experiment, the Overall Evaluation Criterion (OEC) was simple: increasing average revenue per user without degrading key user engagement metrics. Results showed that the newly added site links increased revenue, but also degraded user metrics and Page-Load-Time, likely because of increased vertical space usage. Even offsetting the space by lowering the average number of mainline ads shown per query, this feature improved revenue by tens of millions of dollars per year with neutral user impact, resulting in extremely high ROI (Return-On-Investment).

While the example above is a visual change for monetization, we use controlled experiments for many areas at Bing. Visual changes range from small tweaks like changing colors, to improving search result captions, to bigger changes like adding video to the homepage, and to a complete makeover of Bing's search result page that rolled out in May 2012 and included a new social pane. We also test usability improvements, such as query auto-suggest, "Did you mean," and search history. Backend changes such as relevance rankers, ad optimization, and performance improvements are constantly being experimented with. Finally, we also experiment with changes to sites generating traffic to Bing, such as MSN.

1.2 The Experimentation System

The problem that the Bing Experimentation System addresses is how to guide product development and allow the organization to assess the ROI of projects, leading to a healthy focus on key ideas that move metrics of interest. While there are many ways to design and evaluate products, our choice of controlled experiments for Knowledge Discovery derives from the desire to reliably identify causality with high precision (which features **cause** changes in customer behavior). In the hierarchy of possible designs, controlled experiments are the gold standard in science [17].

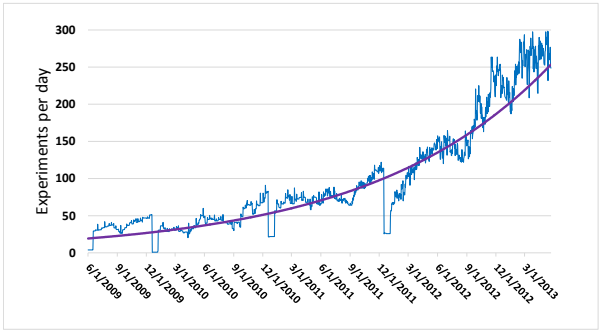

Figure 2: Exponential growth in experimentation over time. (Prior to 2012, Bing shut down most experiments the last two weeks of December.)

If an organization wants to make data-driven decisions to drive product development, with customers' actual behavior as the source of data for decisions, one of the key goals is to enable experimentation at scale: support running many experiments and lower the cost of experimentation. This must be done without lowering the trustworthiness of the overall system.

With the mission of accelerating software innovation through trustworthy experimentation, the use of experimentation at Bing grew exponentially fast over time, as shown in Figure 2.

The Bing Experimentation System is one of the largest systems in the world for running online controlled experiments, with over 200 experiments running concurrently, exposing about 100 million active monthly customers to billions of Bing variants that include implementations of new ideas and variations of existing ones.

1.3 Related Work and Contributions

Multiple papers and books have been written on how to run an online controlled experiment [18; 7; 19; 20] and we will not address that here; we follow the terminology of *Controlled experiments on the web: survey and practical guide* [18]. We build upon that work and share how to scale experimentation, i.e., how to run many experiments to accelerate innovation in product development. We are aware of only one paper that focused on this aspect of experiment scale, an excellent paper by Diane Tang et al. about overlapping experiments at Google [3]. Because that topic is well covered in that paper, and Bing's system is similar [21 pp. 33-34], we chose not to discuss it here. To the best of our knowledge, most of the lessons we share here are novel and not previously covered. Our contributions are as follows:

1. We share key tenets, or principles, which an organization should adopt before using online controlled experiments at scale. Experimentation is not a panacea for everyone, and the assumptions should be understood.
2. We discuss cultural and organizational issues, including two topics not commonly discussed: the cost/benefits of running controlled experiments, and running negative experiments.
3. We discuss engineering challenges, including the system architecture, and alerting, a necessary ingredient when running experiments at scale. We share the results of our study on the impact of the Experimentation System itself.
4. We discuss trustworthiness and statistical challenges above and beyond those usually mentioned as pitfalls for running a single online controlled experiment [22; 23]. In particular, addressing false positives, and both preventing and detecting pairwise interactions.

The lessons we share apply to a wide gamut of companies. Running experiments at large scale does not require a large web site or service: startups have utilized controlled experiments when they have had thousands of active users and are typically looking for large effects. In fact, establishing the experimentation culture early can help startups make the right critical decisions and develop a customer-focused development organization that accelerates innovation [10; 11].

Use of the Bing Experimentation System grew so much because it is credited with having accelerated innovation and increased annual revenues by hundreds of millions of dollars. It allowed us to find and focus on key ideas evaluated through thousands of controlled experiments. The system also helped us identify many negative features that we avoided deploying, despite early excitement by key stakeholders, saving us similar large amounts.

2. TENETS

Running online controlled experiments is not applicable for every organization. We begin with key tenets, or assumptions, an organization needs to adopt.

Tenet 1: The Organization wants to make data-driven decisions and has formalized the Overall Evaluation Criterion (OEC)

You will rarely hear someone at the head of an organization say that they don't want to be data-driven (a notable exception is Apple under Steve Jobs, where Ken Segall claimed that "we didn't test a single ad. Not for print, TV, billboards, the web, retail, or anything" [24 p. 42]). But measuring the incremental benefit to users from new features has costs, and objective measurements typically show that progress is not as rosy as initially envisioned. Many organizations will therefore not spend the resources required to define and measure progress. It is often easier to generate a plan, execute against it, and declare success, with the key metric being: "percent of plan delivered," ignoring whether the feature has any positive impact to key metrics.

In this paper, we assume that the OEC, or Overall Evaluation Criterion, has been defined and can be measured over relatively short durations (e.g., two weeks). In large organizations, it is possible to have multiple OECs, or several key metrics that are shared with refinements for different areas. The hard part is finding metrics that are measurable in the short-term that are predictive of long-term goals. For example, "Profit" is not a good OEC, as short-term theatrics (e.g., raising prices) can increase short-term profit, but hurt it in the long run. As we showed in *Trustworthy Online Controlled Experiments: Five Puzzling Outcomes Explained* [25], market share can be a long-term goal, but it is a terrible short-term criterion: making a search engine worse forces people to issue more queries to find an answer, but, like hiking prices, users will find better alternatives long-term. Sessions per user, or repeat visits, is a much better factor in the OEC, and one that we use at Bing. Thinking of the drivers of lifetime value can lead to a strategically powerful OEC [18]. We cannot overemphasize the importance of coming up with a good OEC that the organization can align behind, but for this paper we will assume this has been done.

Tenet 2: Controlled experiments can be run and their results are trustworthy

Not every decision can be made with the scientific rigor of a controlled experiment. For example, you cannot run a controlled experiment on the possible acquisition of Yahoo! by Microsoft. Hardware devices may have long lead times for manufacturing and modifications are hard, so controlled experiments are hard to run on a new phone or tablet. In *Online Experimentation at Microsoft* [5], the necessary ingredients for running controlled experiments were reviewed. The key point is that for customer-facing web sites, changes are easy to make through software, and running controlled experiments is relatively easy.

Assuming you can run controlled experiments, it is important to ensure their trustworthiness. When running online experiments, getting numbers is easy; getting numbers you can trust is hard, and we have had our share of pitfalls and puzzling results [23; 25; 26]. For this paper, we assume the organization does the checks for correctness and is aware of the pitfalls.

Tenet 3: We are poor at assessing the value of ideas

Features are built because teams believe they are useful, yet in many domains most ideas fail to improve key metrics. Only one third of the ideas tested at Microsoft improved the metric(s) they were designed to improve [5]. Success is even harder to find in well-optimized domains like Bing. Jim Manzi [17] wrote that at Google, only "about 10 percent of these [controlled experiments, were] leading to business changes." Avinash Kaushik wrote in his Experimentation and Testing primer [27] that "80% of the time you/we are wrong about what a customer wants." Mike Moran [28 p. 240] wrote that Netflix considers 90% of what they try to be wrong. Regis Hadiaris from Quicken Loans wrote that "in the five years I've been running tests, I'm only about as correct in guessing the results as a major league baseball player is in hitting the ball. That's right - I've been doing this for 5 years, and I can only "guess" the outcome of a test about 33% of the time!" [4]. Dan McKinley at Etsy wrote [29] "nearly everything fails" and "it's been humbling to realize how rare it is for them [features] to succeed on the first attempt. I strongly suspect that this experience is universal, but it is not universally recognized or acknowledged." Finally, Colin McFarland wrote in the book *Experiment!* [7 p. 20] "No matter how much you think it's a no-brainer, how much research you've done, or how many competitors are doing it, sometimes, more often than you might think, experiment ideas simply fail."

Not every domain has such poor statistics, but most who have run controlled experiments in customer-facing web sites and applications have experienced this humbling reality: we are poor at assessing the value of ideas.

3. CULTURAL AND ORGANIZATIONAL LESSONS

We now discuss areas related to cultural and organizational aspects.

3.1 Why Controlled Experiments?

The most common question we get as an organization learns about controlled experiments is "why not measure the metric of interest, ship the feature, and then look at the delta?" Alternatively, look at correlations with metrics of interest.

Our experience is that external variations overwhelm the effects we are trying to detect [23]. In sequential tests, or quasi-experimental designs, we try to control for known confounding factors, but this is extremely hard to get right. As the common proverb goes: correlation is not causation. Here are the best examples we found to drive the point across

1. Stanley Young and Alan Karr [30] compared published results from medical hypotheses shown to be significant using observational studies with randomized clinical trials, considered more reliable. Their conclusion: "Any claim coming from an observational study is most likely to be wrong."

2. Ioannidis's papers [31; 32] (the first is *the* most downloaded technical paper in the Public Library of Science Medicine journal) showed that uncontrolled / nonrandomized experiments have a much higher probability of being false. Manzi [17 p. 91] summarizes the papers as follows: "[Ioannidis] evaluated the reliability of forty-nine influential studies (each cited more than 1,000 times) published in major journals between 1990 and 2003 that reported effective interventions based on either experimental or non-experimental methods…90 percent of large randomized experiments produced results that stood up to replication, as compared to only 20 percent of nonrandomized studies." While the numbers are very small, these are convincing papers.

Additional accessible stories in the popular press have been very convincing [33; 34].

Our experience is that organizations go through four stages as they learn to experiment [35]: (1) **Hubris**, where measurement is not needed because of confidence in the HiPPO (Highest Paid Person's Opinion). (2) **Measurement and Control**, where the organization measures key metrics and starts to control for unexplained differences. As Thomas Kuhn notes, paradigm shifts happen "only through something's first going wrong with normal research" [36]. (3) **Semmelweis Reflex** [37], where the organization rejects new knowledge because it contradicts entrenched norms, beliefs or paradigms. (4) **Fundamental understanding**, where causes are understood and models actually work.

3.2 Cost vs. Benefit and the Ideas Funnel

There are many methods that can be used to evaluate new ideas, including pitching the ideas to others; reviewing sketches, mockups, and prototypes; conducting surveys and usability lab studies; tests against historical data; and running controlled experiments. These evaluation methods vary both in the cost to execute them as well as the value and reliability of the information gained through them. In *How to Measure Anything: Finding the Value of Intangibles in Business* [38], Doug Hubbard used the term EVI, Expected Value of Information, to define the expected benefit gained by getting additional information. A controlled experiment provides very close to perfect information (up to the uncertainty from the p-value and other experimental design factors), but it can be more expensive than other methods of evaluating new ideas.

Organizations should consider a large number of initial ideas and have an efficient and reliable mechanism to narrow them down to a much smaller number of ideas that are ultimately implemented and released to users in online controlled experiments. For this funnel of ideas to be efficient, low cost methods such as pitching ideas and reviewing mockups are needed to evaluate and narrow down the large number of ideas at the top of the funnel. Controlled experiments are typically not suitable to evaluate ideas at the top of the funnel because they require each idea to be implemented sufficiently well to deploy and run on real users, and this feature development cost can be high. Hence, at the top of the funnel more ideas are evaluated using low-cost techniques, but with lower fidelity. Conversely, at the bottom of the funnel there are fewer ideas to evaluate and the organization should use more reliable methods to evaluate them, with controlled experiments being the most reliable and preferred method.

A key observation is that if a controlled experiment is cheap to run, then other evaluation methods rarely make sense. For example, some ideas are easy to code and deploy; other involve changing configuration parameters. One reason for using other methods in these cases is to gain qualitative feedback (e.g., through surveys and usability lab studies); however these other methods should be used to complement controlled experiments and not to replace them since the quantitative information they provide is inferior.

3.3 Test Everything in Experiments

In the previous section, we focused on evaluating new ideas. But what about platform changes, code refactoring, and bug fixes?

In a platform change, you replace an underlying platform component with a new-and-better version. The team responsible for the new platform component claims that the new one is faster, takes up less memory, and does everything with a new-and-better code base that's easier to maintain, faster to innovate, and fully tested for compatibility. They've been working on it for six months, passed all exit criteria, and are ready to deploy. In a data-driven org, the final test has to be a controlled experiment: run the new component in an A/B test and see that you get no significant differences (or even better, some improvements). The reality is that the new code typically does not handle the edge cases as well as the old code, and it is very likely more buggy. The first author remembers how the Amazon Order Pipeline team wanted to introduce the new version based on the new app server, Gurupa, and he insisted that an A/B test be run: it failed with a 2% revenue loss. The team dug deep for two weeks, found "the" bug, and wanted to ship. No, you need to pass an A/B test was the message. The team ran it and it failed again. The new pipeline shipped after five iterations. It is not just new ideas that fail, but re-implementations of existing ones are not as good as we initially think.

Code refactoring and bug fixes present an interesting tradeoff. For a large organization, there are many small fixes that go in every day, and it would be unreasonable to run controlled experiments for each one. We recommend that small fixes get bundled into packages so that if one is egregiously bad, the package will test negatively and it will be identified. Building the infrastructure to do this cheaply and efficiently is the real challenge.

The key is to admit that mistakes will happen and try to run controlled experiments to detect them. A bug that introduces a 1% reduction to revenue costs Bing over $10M per year in the US alone. Detecting the 1% is easy in a controlled experiment; it is much harder based on sequential patterns.

3.4 Negative Experiments

The question of whether we should run controlled experiments that knowingly degrade the user experience (e.g., slowing performance) is highly polarizing. Some people believe we should never knowingly degrade the user experience. Over time, we achieved agreement that knowingly hurting users in the short-term (e.g., a 2-week experiment) can let us understand fundamental issues and thereby improve the experience long-term. We believe that this is not only justified, but should be encouraged. The Hippocratic Oath is often associated with the phrase "Do no harm" (although not precisely phrased that way), yet there is strong evidence that doctors have been harming patients for millennia. In *Bad Medicine: Doctors Doing Harm Since Hippocrates*, David Wootton [39] wrote that "For 2,400 years patients have believed that doctors were doing them good; for 2,300 years they were wrong." Doctors did bloodletting for hundreds of years, thinking it had a positive effect, not realizing that the calming effect was a side effect that was unrelated to the disease itself. When President George Washington was sick, doctors extracted about 35%-50% of his blood over a short period, which inevitably led to preterminal anemia, hypovolemia, and hypotension. The fact that he stopped struggling and appeared physically calm shortly before his death was probably due to profound hypotension and shock. Running control

experiments on changes that we believe are "negative" to confirm the causal effect is critical so that we do not make the same mistake doctors did for centuries. Even if the HiPPO (Highest Paid Person's Opinion) in your organization is strongly held, we recommend validating it. Hippocrates' "Do no harm" should really be "Do no long-term harm."

Understanding the impact of performance (speed) on key metrics is a fundamental question. There is an interest in isolating the performance and answering: excluding the impact due to performance (typically degradations), did my feature improve some key metrics? Initial implementations are often slow and if one is building a Minimum-Viable-Product (MVP) to test an idea, it is best not to start optimizing performance before validating that the idea itself is good. Quantifying the relationship between changes in performance and changes to key metrics is highly beneficial to the organization.

This quantification may change over time, as the site's performance and bandwidth standards improve. Bing's server performance is now sub-second at the 95th percentile. Past experiments showed that performance matters, but is it still the case? We recently ran a slowdown experiment where we slowed 10% of users by 100msec (milliseconds) and another 10% by 250msec for two weeks. The results showed that performance absolutely matters a lot today: every 100msec improves revenue by 0.6%. The following phrasing resonated extremely well in our organization (based on translating the above to profit): *an engineer that improves server performance by 10msec (that's 1/30 of the speed that our eyes blink) more than pays for his fully-loaded annual costs*. Every millisecond counts.

3.5 Beware of Twyman's Law and Stories in the Wild

Twyman wrote that "*Any figure that looks interesting or different is usually wrong."* We recommend healthy skepticism towards stories depicting astounding results from tiny changes, such as 50% revenue lift due to changing the color of the Buy Button. While we have some unexpected successes from small changes, they are extremely rare. Most amazing results turn out to be false when reviewed carefully [23; 25], so they need to be replicated with high statistical power and deeply analyzed before we believe them.

Some sites, such as http://whichtestwon.com, share the test of the week. Our experience is that there are good ideas and hypotheses that are worth evaluating, but Ioannidis' warnings [31] apply well here: we suspect many results are phrased too strongly or are incorrect. Multiple-testing, bias, and weak standards lower the trust one should have in these results (e.g., the test at http://whichtestwon.com/whichtestwons-overlay-timer-test was published based on a non-stat-sig p-value, >0.05). Our recommendation is classical science: replication. If you find a great hypothesis, retest it on your site.

We want to share one example where we found a result contradicting ours. In Section 3.4, we mentioned that performance matters a lot; Greg Linden [40 p. 15] noted that 100msec slowdown at Amazon impacted revenue by 1%; a paper by co-authors from Bing and Google [41] showed the significant impact of performance on key metrics. With so much evidence, we were surprised to see Etsy's Dan McKinley [2] claim that a 200msec delay did not matter. It is possible that for Etsy users, performance is not critical, but we believe a more likely hypothesis is that the experiment did not have sufficient statistical power to detect the differences. Clearly if you increase the slowdown it will matter at some point: at 5 minutes, there will be close to zero engagement, so where on the continuum can Etsy detect the impact? 500msec? One second? Telling an organization that performance doesn't matter will make the site slower very quickly, to the point where users will abandon in droves. We believe Etsy should either increase statistical power, or increase the delay until they are able to get a statistically significant signal, and they might be surprised by the impact on their key metrics.

3.6 Innovation vs. Incrementalism: 41 Shades of Blue

As experimentation becomes "low cost," it is easy to fall into the trap of answering many trivial questions by running controlled experiments. This is well exemplified in Douglas Bowman's blog [42], describing how a team at Google that couldn't agree on a blue color for a link experimented with 41 shades of blue. While such variations could be important in some cases, many make no difference and may discourage thoughtful designs.

Experimentation is a tool, and we agree that it can support a quick "try, evaluate, ship" cycle that provides the illusion of progress if the steps are tiny: you don't get to the moon by climbing higher and higher trees. Conversely, we have seen big bets that could have been declared a big success by the HiPPO, were it not for the fact that controlled experiments provided objective judgment that key metrics did not really move. As with any funnel of ideas, one must evaluate the total benefit of several small incremental bets vs. some big bold risky bets. As with stocks, an organization is usually better with a portfolio of ideas at different points on the risk/reward curve.

Sometimes an organization has to take a big leap in the space of options and start to hill-climb in a new area in order to see if it is near a taller mountain. The initial jump might end up lower than the current local maxima, and it may take time to explore the new area. As the initial explorations fail to beat the current champion, the question of "fail fast" vs. "persevere" always comes up. There is no magic bullet here: it is about running some experiments to get a sense of the "terrain" and being open to both options.

3.7 Multivariate Tests

Multivariate Tests (MVT) evaluate the impact of multiple variables that could interact, and are the subject of a rich statistical literature [20] and many buzzword-compliant brochures of product vendors. We have previously made the case that in the online world, agility and continuous availability of users makes MVTs less appealing [18]. Researchers at Google made similar observations [3]. Despite the massive growth in experimentation, we continue to believe that the current orthogonal design (equivalent to a full-factorial) is the most appropriate. In our experience, interactions are relatively rare and more often represent bugs than true statistical interactions (also see Section 5.2). When we do suspect interactions, or when they are detected, we run small MVTs, but these are relatively rare.

4. ENGINEERING LESSONS

As the Bing organization has embraced controlled experiments for decision making, there is a continuing need to scale the platform for running experiments while lowering the per-experiment costs, and keeping the trust level high by making it hard for the uninformed to make mistakes. Key to this scaling is an investment in self-service tools for creating, managing and analyzing experiments. These guided tools enable all engineers in the organization to run their own experiments and act on the outcomes. While a small centralized team creates these tools and provides guidance and oversight to encourage high quality experimentation, the decision making over what experiments to run and how to incorporate the feedback is largely decentralized. Different aspects of the system are monitored and tuned to address scaling

challenges, especially the use of limited resources, such as users to allocate to tests and machines for analysis.

4.1 Architecture

Bing's experiment system architecture is outlined in Figure 3, and covers four key areas. For this section, we use Bing's terminology. A *flight* is a variant that a user is exposed to. A *Number Line* is an orthogonal assignment, similar to Google's layers [3]. This mechanism provides guaranteed isolation from conflicting assignment: a user will be in only one flight per number line. A user is assigned to multiple concurrent flights, one per number line. The four key areas of the architecture are:

1. **Online Infrastructure**. As a request is received from a browser, Bing's frontend servers assign each request to multiple *flights* running on a set of number lines. To ensure the assignment is consistent, a pseudo random hash of an anonymous user id is used [18]. The assignment happens as soon as the request is received and the frontend then passes each request's flight assignments as part of the requests sent to lower layers of the system. All systems in Bing are driven from configuration and an experiment is implemented as a change to the default configuration for one or more components. Each layer in the system logs information, including the request's flight assignments, to system logs that are then processed and used for offline analysis.
2. **Experiment Management**. Experimenters use a system, called *Control Tower*, to manage their experiments. To support greater automation and scale, all tools, including Control Tower, run on top of APIs for defining and executing experiments and experiment analysis. Some groups build on these APIs to automate experiment execution (e.g. for large scale automated experimentation). A configuration API and tool enables experimenters to easily create the setting defining an experiment.
3. **Offline Analysis**. An experiment summary is called a *scorecard*, and is generated by an offline experiment analysis pipeline that must manage a large scale analysis workload–both in data and volume of experiments. Using the logs, the system manages and optimizes the execution of multiple analysis workflows used for experiment scorecards, monitoring and alerting, as well as deep dive analytics. The scorecards enable simple slicing and dicing as well as viewing the changing impact of an experiment over time. An alerting and monitoring system automatically detects both data quality and adverse user impact events, as described in Section 4.3.

4.2 Impact of the Experimentation System

Although experimentation is critical for data driven product innovation, it does not come without cost. To the best of our knowledge, these costs have never been documented in detail. In this section we describe how we evaluated the impact of the experimentation system itself, including the average impact from live experiments over several months.

As discussed in Section 4.1, the Experimentation System affects all layers of the system and has a performance impact at each layer. First, the experiment assignment adds a small delay (less than a millisecond). Second, increasing the number of experiments assigned to each request results in increasing cache fragmentation, lowering cache hit rates and increasing latency. Bing caches the first n results for common queries, but treatments cannot share a cache entry if they return different results for the same request. As the number of concurrent experiments that influence search results increases, fragmentation increases exponentially. For example, our ranker has four layers, and if three treatments (+ 1 control) are running concurrently (on different number lines), we fragment the cache by a factor of 4^4 = 256! Finally, new features are typically less performance-optimized in early incarnations.

To quantify the impact of Bing's Experimentation System, we holdout 10% of our total users from any experimentation. This holdout group serves as a "top-level control" while the rest of the users are in experimental treatments. In short, the problem of understanding the impact of Experimentation System itself becomes another A/B test (we ignore the assignment to the holdout group, as it is extremely fast). We monitor key metrics continuously, and take action if we find the impact exceeds a prescribed budget level.

We quantified the impact of the Experimentation System on multiple metrics internally, and we share one here: speed, or page-load-time. In Bing, a key performance metric is Page-Load-Time, which is defined as the time from the user's query to the browser's firing of the onload event on the resulting page. The experiment group consistently showed a 25msec to 30msec delay. A separate study for the impact of web cache shows the web cache fragmentation alone contributes about 20msec. It is clear that by doing experimentation, there is a cost of learning and we believe

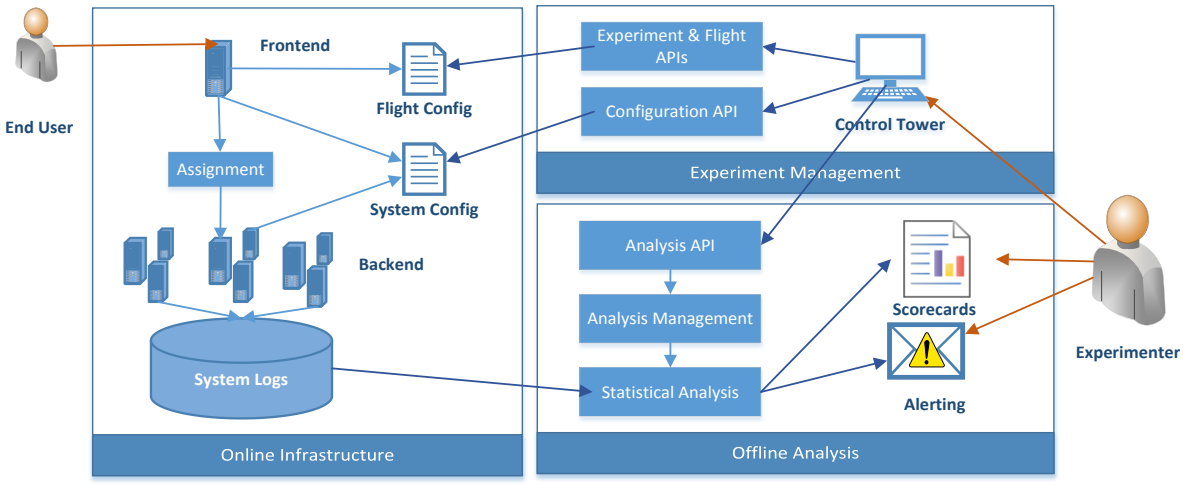

Figure 3: Experimentation System Architecture

being able to quantify the cost is important for any organization that runs experiments at large scale.

4.3 Alerts and Aborting Bad Experiments

Any change has the potential to degrade the user experience, and even a small degradation can increase user abandonment or cost millions of dollars if not caught quickly. As the organization grows, and number and frequency of feature changes increases, so does the need to automatically detect and alert when degradations occur and to automatically revert any changes that cause severe degradations. This use of controlled experiments provides a critical safety net which enables a company to scale the number of ideas tested and changes made while still maintaining a tolerable level of risk.

The naïve approach to alerting on any statistically significant negative metric changes will lead to an unacceptable number of false alerts and thus make the entire alerting system useless. To avoid this we employ multiple techniques:

1. Before raising an alert, we require that a detected delta is not only statistically significant but also large enough in absolute magnitude to have meaningful user or business impact. For example, we do not alert on a 2 millisecond degradation to Page-Load-Time, even if we have very high confidence the degradation is a real effect (e.g., p-value less than 1e-10).
2. Corrections for multiple testing. The O'Brien & Fleming procedure [43] calls for lower p-values early on and these increase over time, as shown in Figure 4. For example, in a 7-day experiment, the p-value cutoff for the 1st day is 5×10^{-8}, which is much smaller than 0.05, while the last cutoff is 0.040. This works well, as earlier termination needs to meet a higher bar, which aligns well with our intuition. Second, the p-value cutoff at the final check point is not much lower than 0.05. This implies that an experiment that is significant under the one-stop testing is likely to be significant under the O'Brien-Fleming as well, while if the results are extreme we gain the benefit of stopping early.
3. Different magnitudes of changes for different metrics are categorized in specific severity levels. The most severe changes result in automatic shutdown of an experiment but less severe changes will result in emails sent to the owner of the experiment and a central experimentation team.

In addition to looking at user and business impact metrics, it is critical to monitor data quality metrics. See Section 8 of the *Seven Pitfalls* paper [22] for recommended audits.

5. TRUSTWORTHINESS and STATISTICAL LESSONS

It is critical that the results of experiments be trustworthy: incorrect results may cause bad ideas to be deployed or good ideas to be incorrectly ruled out. With a large system, false positives are inevitable, so we try to minimize their impact. As a user is put into more and more concurrent experiments, the chance of unexpected interactions between those experiment increases, which can lead to misleading results, and hinder scaling. Preventing interactions where possible, and detecting where not, has been a critical element for delivering trustworthy, large scale experimentation.

5.1 False Positives

False positives are "positive findings" that are not actually true. They can be due to experimental design issues, data issues, biased analyses, or simply chance. It is known that causal inferences using observational data have much higher false positive rates than a proper conducted controlled experiment [17]. But as Ioannidis showed [31], false positives in controlled experiments can still be

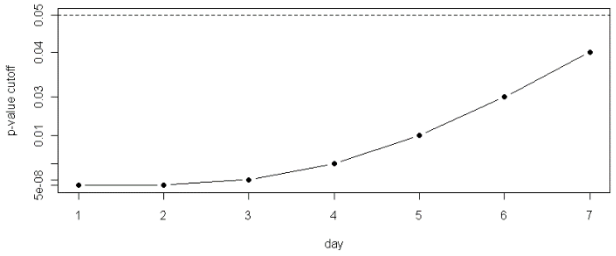

Figure 4: O'Brien-Fleming p-value thresholds as the experiment progresses, with 7 check points

higher than we expect. To avoid the design and analysis biases described by Ioannidis, we standardized our designs and automated experiment analyses.

When statistical hypothesis testing is executed properly, the false positive rate is controlled by the p-value threshold, usually set at 5%. This rate assumes one data set, one outcome, and one analysis. In practice, we violate each of those assumptions.

First, in online experimentation, data are collected sequentially. If we check the results every day, then the one dataset assumption is violated and we are exposed to false positives due to multiple testing (also see Section 4.3). While we allow experimenters to look at daily results, as they lead to insights and could help identify bugs early on, there is one final scorecard at the end of the experiment, which we require to be a multiple of weeks, usually two weeks.

Second, we report results on not one metric, but on hundreds, mostly to aid debugging and analysis. To address the multiple outcomes issue, we standardized our success criteria to use a small set of metrics, such as sessions/user [25].

Third, we violate the one analysis assumption as experimenters slice and dice the data many ways– e.g. to understand the impact on specific user segments like one type of browser. For multiple analyses (slice-and-dice), we educate experimenters on false positives, and encourage them to adjust their probability threshold, focusing on strong signals and smaller p-values (e.g., < 1e-4).

As teams iteratively improve a feature based on experiment results, a new idea may go through a sequence of tens of controlled experiments. The risk is that a team may get a significant result by chance, celebrate, and ship. Assuming the feature does nothing, running k iterations (each with small variations that do nothing), then the probability of statistical significance grows from 2.5% (positive movement in a two-sided test) to $(1 - 0.975^k)$. The problem is exacerbated when teams run multiple treatments. If a team tries five treatments, then the 2.5% false positive rate grows to 12%. If they do six iterations of 5-treatment experiments, there is more than a 50% chance of getting a positive statistically significant result. Two mechanisms are used to protect us from these false positives:

1. **Threshold adjustments**. We look for lower p-values for projects that have multiple treatments and/or iterations.
2. **Replication Stage**. While we encourage trying multiple variants, once the funnel narrows, there should be a "final" run, preferably with higher statistical power, which determines the final results.

In any large scale system false positives are a commonplace given the many ways in which we violate the assumption of a single hypothesis test done once. Our approach is pragmatic: we accept that fact and adjust our practices to reduce the rate of occurrence (e.g. by adjusting the threshold) and use replication as the final

check to avoid false positives and get a more accurate (unbiased) estimate of the effect size.

5.2 Interaction Prevention and Detection

As we increase the number of experiments running in parallel, the risk of interactions between different treatments becomes a growing concern. A *statistical interaction* between two treatments A and B exists if their combined effect is not the same as the sum of two individual treatment effects [18]. The existence of interaction violates the basic premise we use to scale experimentation: that each experiment can be analyzed in isolation. In an organization running hundreds of experiments daily, interactions pose a serious threat to experiment trustworthiness. First, interactions can harm users, because particular combinations can trigger unexpected bugs and cause a negative user experience. Second, interactions skew experiment results for all experiments involved. This is extremely important when the real treatment effect is small, as a small interaction can give completely misleading results for a key metric. Finally, it is impossible to completely prevent interaction in a large-scale experimentation system through testing and other offline checks. Different teams focusing on their own area do not know or check interactions with features tested by other teams.

A comprehensive solution for interaction includes both prevention and detection. To prevent interactions, each Bing experiment defines a set of constraints. The experiment system uses those constraints to ensure that conflicting experiment do not run together. For example, a constraint associated with all ad visual experiments ensures that a user is never assigned to two such experiments at the same time. Another key tool for prevention uses a configuration management system to automatically detect experiments trying to change the same configuration parameter prior to launch [3]. Finally, when interactions cannot be avoided we use "mappings" to exclude users in one experiment from appearing in another experiment.

Prevention is never perfect and it is critical to detect what we cannot prevent. Interaction detection for a large scale system is itself a problem of scale: If we are running N experiments at a time, the complexity of detecting pairwise-interactions is quadratic in N. Bing's Experimentation System monitors all running experiments for potential pairwise interactions on a set of metrics, both key user metrics as well a set of metrics we have found sensitive to interactions. Because of the large scale of experimentation, the system must sometimes run hundreds of thousands of hypothesis tests. To prevent a high false positives rate we use an Empirical Bayesian False Discovery Rate control algorithm to identify cases that are most likely be true positive [44]. After detecting an interaction, the tool will automatically run a deeper analysis and diagnose the most important interactions, which are sent as an alert to the experiment owners.

6. CONCLUSION

Anyone who has been running online controlled experiments knows how humbling it is to get an objective assessment of your ideas by real users. After an initial period of disappointment that our "gut" feelings and intuition mislead us so often, one recognizes that the ability to separate the truly good ideas from the rest is an innovation accelerator and a core organizational competency.

We shared the challenges and lessons in scaling to run a large number of experiments. We covered three broad areas: cultural / organizational, engineering, and trustworthiness, and covered issues including cost/benefit tradeoffs, negative experiments, incrementalism concerns, dealing with false positives and interactions, and an evaluation of the overall impact of the experimentation system itself.

One aspect of scaling that we did not discuss is scaling through improved sensitivity. Better sensitivity is crucial in scaling an experiment system, as it effectively increases the number of experiments that can run concurrently without requiring more users. In Bing, we started using pre-experiment data to reduce the variance and improve sensitivity [45], but we believe there is room for significant improvements in this area.

We hope these lessons will allow others to scale their systems and accelerate innovation through trustworthy experimentation.

ACKNOWLEDGMENTS

We wish to thank Xavier Amatriain, Steve Blank, Seth Eliot, Juan Lavista Ferres, Yan Guo, Greg Linden, Yoelle Maarek, Llew Mason, and Dan McKinley for their feedback. We have been fortunate to have been part of Bing during the massive growth in experimentation, and wish to thank many people for encouraging data-driven decision making, especially Qi Lu and Harry Shum.

References

1. **Kohavi, Ron and Round, Matt.** *Front Line Internet Analytics at Amazon.com.* [ed.] Jim Sterne. Santa Barbara, CA : s.n., 2004. http://ai.stanford.edu/~ronnyk/emetricsAmazon.pdf.

2. **McKinley, Dan.** Design for Continuous Experimentation: Talk and Slides. [Online] Dec 22, 2012. http://mcfunley.com/design-for-continuous-experimentation.

3. **Tang, Diane, et al.** Overlapping Experiment Infrastructure: More, Better, Faster Experimentation. *Proceedings 16th Conference on Knowledge Discovery and Data Mining.* 2010.

4. **Moran, Mike.** Multivariate Testing in Action: Quicken Loan's Regis Hadiaris on multivariate testing. *Biznology Blog by Mike Moran.* [Online] December 2008. www.biznology.com/2008/12/multivariate_testing_in_action/.

5. **Kohavi, Ron, Crook, Thomas and Longbotham, Roger.** Online Experimentation at Microsoft. *Third Workshop on Data Mining Case Studies and Practice Prize.* 2009. http://exp-platform.com/expMicrosoft.aspx.

6. **Amatriain, Xavier and Basilico , Justin.** Netflix Recommendations: Beyond the 5 stars. [Online] April 2012. http://techblog.netflix.com/2012/04/netflix-recommendations-beyond-5-stars.html.

7. **McFarland, Colin.** *Experiment!: Website conversion rate optimization with A/B and multivariate testing.* s.l. : New Riders, 2012. 978-0321834607.

8. **Kolar, Sumanth.** Recommendations and Discovery at StumbleUpon. [Online] Sept 2012. www.slideshare.net/sumanthkolar/recsys-2012-sumanth-14260370.

9. **Smietana, Brandon.** Zynga: What is Zynga's core competency? *Quora.* [Online] Sept 2010. http://www.quora.com/Zynga/What-is-Zyngas-core-competency/answer/Brandon-Smietana.

10. **Blank, Steven Gary.** *The Four Steps to the Epiphany: Successful Strategies for Products that Win* . s.l. : Cafepress.com, 2005. 978-0976470700.

11. **Ries, Eric.** *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses* . s.l. : Crown Business, 2011. 978-0307887894.

12. **Wikipedia.** Lean Startup. [Online] 2013. http://en.wikipedia.org/wiki/Lean_Startup.

13. **Sinofsky, Steven.** *One Strategy: Organization, Planning, and Decision Making* . s.l. : Wiley, 2009. 978-0470560457 .

14. **comScore.** comScore Releases January 2013 U.S. Search Engine Rankings. [Online] Feb 13, 2013. http://www.comscore.com/Insights/Press_Releases/2013/2/comScore_Releases_January_2013_U.S._Search_Engine_Rankings.

15. *SCOPE: Parallel Databases Meet MapReduce.* **Zhou, Jingren, et al.** s.l. : VLDB Journal, 2012. http://research.microsoft.com/en-us/um/people/jrzhou/pub/Scope-VLDBJ.pdf.

16. **Klein, Peter and Suh, Chris.** Microsoft Second Quarter 2013 Earnings Calls Transcript. *Microsoft Investor Relations.* [Online] Jan 24, 2013. http://www.microsoft.com/global/Investor/RenderingAssets/Downloads/FY13/Q2/Microsoft_Q2_2013_PreparedRemarks.docx.

17. **Manzi, Jim.** *Uncontrolled: The Surprising Payoff of Trial-and-Error for Business, Politics, and Society.* s.l. : Basic Books, 2012. 978-0-465-02931-0.

18. **Kohavi, Ron, et al.** Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery.* February 2009, Vol. 18, 1, pp. 140-181. http://www.exp-platform.com/Pages/hippo_long.aspx.

19. **Eisenberg, Bryan.** How to Improve A/B Testing. *ClickZ Network.* [Online] April 29, 2005. www.clickz.com/clickz/column/1717234/how-improve-a-b-testing.

20. **Box, George E.P., Hunter, J Stuart and Hunter, William G.** *Statistics for Experimenters: Design, Innovation, and Discovery.* 2nd. s.l. : John Wiley & Sons, Inc, 2005. 0471718130.

21. **Kohavi, Ron.** Online Controlled Experiments: Introduction, Learnings, and Humbling Statistics. *The ACM Conference on Recommender Systems.* 2012. Industry Keynote. http://www.exp-platform.com/Pages/2012RecSys.aspx.

22. **Crook, Thomas, et al.** Seven Pitfalls to Avoid when Running Controlled Experiments on the Web. [ed.] Peter Flach and Mohammed Zaki. *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining.* 2009, pp. 1105-1114. http://www.exp-platform.com/Pages/ExPpitfalls.aspx.

23. **Kohavi, Ron and Longbotham, Roger.** Unexpected Results in Online Controlled Experiments. *SIGKDD Explorations.* 2010, Vol. 12, 2. http://www.exp-platform.com/Documents/2010-12%20ExPUnexpectedSIGKDD.pdf.

24. **Segall, Ken.** *Insanely Simple: The Obsession That Drives Apple's Success.* s.l. : Portfolio Hardcover, 2012. 978-1591844839.

25. **Kohavi, Ron, et al.** Trustworthy online controlled experiments: Five puzzling outcomes explained. *Proceedings of the 18th Conference on Knowledge Discovery and Data Mining.* 2012, www.exp-platform.com/Pages/PuzzingOutcomesExplained.aspx.

26. **Kohavi, Ron, Longbotham, Roger and Walker, Toby.** Online Experiments: Practical Lessons. [ed.] Simon S.Y. Shim. *IEEE Computer.* September 2010, Vol. 43, 9, pp. 82-85. http://www.exp-platform.com/Documents/IEEE2010ExP.pdf.

27. **Kaushik, Avinash.** Experimentation and Testing: A Primer. *Occam's Razor.* [Online] May 22, 2006. http://www.kaushik.net/avinash/2006/05/experimentation-and-testing-a-primer.html.

28. **Moran, Mike.** *Do It Wrong Quickly: How the Web Changes the Old Marketing Rules* . s.l. : IBM Press, 2007. 0132255960.

29. **McKinley, Dan.** Testing to Cull the Living Flower. [Online] Jan 2013. http://mcfunley.com/testing-to-cull-the-living-flower.

30. *Deming, data and observational studies: A process out of control and needing fixing.* **Young , S Stanley and Karr, Allan.** 3, 2011, Significance, Vol. 8. http://www.niss.org/sites/default/files/Young%20Karr%20Obs%20Study%20Problem.pdf.

31. *Why Most Published Research Findings Are False.* **Ioannidis, John P.** 8, 2005, PLoS Medicine, Vol. 2, p. e124. http://www.plosmedicine.org/article/info:doi/10.1371/journal.pmed.0020124.

32. *Contradicted and Initially Stronger Effects in Highly Cited Clinical Research.* **Ioannidis, John P.** 2, s.l. : The Journal of the American Medical Association, 2005, Vol. 294. http://jama.jamanetwork.com/article.aspx?articleid=201218.

33. **Weiss, Rick.** Study Debunks Theory On Teen Sex, Delinquency. *Washington Post.* 2007, Nov 11. http://www.washingtonpost.com/wp-dyn/content/story/2007/11/11/ST2007111100542.html.

34. **Myopia, How Science Works: The Story of Night-Light.** Myopia: Prevention and Control. [Online] http://www.myopiaprevention.org/references_night_lights.html.

35. **Kohavi, Ron.** Online Controlled Experiments: Listening to the Customers, not to the HiPPO. *Keynote at EC10: the 11th ACM Conference on Electronic Commerce.* 2010. http://www.exp-platform.com/Documents/2010-06%20EC10.pptx.

36. **Kuhn, Thomas.** *The Structure of Scientific Revolutions.* 3rd. 1996. 978-0226458083 .

37. **Wikipedia.** *Semmelweis reflex.* http://en.wikipedia.org/wiki/Semmelweis_reflex.

38. **Hubbard, Douglas W.** *How to Measure Anything: Finding the Value of Intangibles in Business.* 2nd. s.l. : Wiley, 2010.

39. **Wooton, David.** *Bad Medicine: Doctors Doing Harm Since Hippocrates.* s.l. : Oxford University Press, 2007.

40. **Linden, Greg.** Make Data Useful. [Online] Dec 2006. home.blarg.net/~glinden/StanfordDataMining.2006-11-29.ppt.

41. *Performance Related Changes and their User Impact.* **Schurman, Eric and Brutlag, Jake.** s.l. : Velocity 09: Velocity Web Performance and Operations Conference, 2009.

42. **Douglas Bowman.** Goodbye, Google. *StopDesign.* [Online] http://stopdesign.com/archive/2009/03/20/goodbye-google.html.

43. *A Multiple Testing Procedure for Clinical Trials.* **O'Brien, Peter C. and Fleming, Thomas R.** 3, September 1979, Biometrics, Vol. 35, pp. 549-556.

44. *A unified approach to false discovery rate estimation.* **Strimmer, Korbinian.** 1, s.l. : Bmc Bioinformatics, 2008, Vol. 9.

45. **Deng, Alex, et al.** Improving the Sensitivity of Online Controlled Experiments by Utilizing Pre-Experiment Data. *WSDM 2013: Sixth ACM International Conference on Web Search and Data Mining.* 2013. www.exp-platform.com/Pages/CUPED.aspx.