This is a public version of a Microsoft ThinkWeek paper that was recognized as top-30 in late 2009

Online Experimentation at Microsoft

Ronny Kohavi
ronnyk@microsoft.com

Thomas Crook
tcrook@microsoft.com

Roger Longbotham
rogerlon@microsoft.com

Brian Frasca
brianfra@microsoft.com

Randy Henne
rhenne@microsoft.com

Juan Lavista Ferres
jlavista@microsoft.com

Tamir Melamed
tamirme@microsoft.com

Controlled experiments, also called randomized experiments and A/B tests, have had a profound influence on multiple fields, including medicine, agriculture, manufacturing, and advertising. Through randomization and proper design, experiments allow establishing causality scientifically, which is why they are the gold standard in drug tests. In software development, multiple techniques are used to define product requirements; controlled experiments provide a valuable way to assess the impact of new features on customer behavior. At Microsoft, we have built the capability for running controlled experiments on web sites and services, thus enabling a more scientific approach to evaluating ideas at different stages of the planning process. In our previous papers, we did not have good examples of controlled experiments at Microsoft; now we do! The humbling results we share bring to question whether a-priori prioritization is as good as most people believe it is. The Experimentation Platform (ExP) was built to accelerate innovation through trustworthy experimentation. Along the way, we had to tackle both technical and cultural challenges and we provided software developers, program managers, and designers the benefit of an unbiased ear to listen to their customers and make data-driven decisions. A technical survey of the literature on controlled experiments was recently published by us in a journal (Kohavi, Longbotham, Sommerfield, & Henne, 2009). The goal of this paper is to share lessons and challenges focused more on the cultural aspects and the value of controlled experiments.

1. Introduction

We should use the A/B testing methodology
a LOT more than we do today

-- Bill Gates, 2008
Feedback to prior Thinkweek paper

On Oct 28, 2005, Ray Ozzie, Microsoft's Chief Technical Officer at the time, wrote *The Internet Services Disruption* memo (Ray Ozzie, 2005). The memo emphasized three key tenets that were driving a fundamental shift in the landscape: (i) The power of the advertising-supported economic model; (ii) the effectiveness of a new delivery and adoption model (discover, learn, try, buy, recommend); and (iii) the demand for compelling, integrated user experiences that "just work." Ray wrote that the "web is fundamentally a self-service environment, and it is critical to design websites and product 'landing pages' with sophisticated closed-loop measurement and feedback systems… This ensures that the most effective website designs will be selected…" Several months after the memo, the first author of this paper, Ronny Kohavi, proposed building an Experimentation Platform at Microsoft. The platform would enable product teams to run controlled experiments.

The goal of this paper is to not to share technical aspects of controlled experiments—we published these separately (Kohavi, Longbotham, Sommerfield, & Henne, 2009)—rather the paper covers the following.

1. **Challenges and Lessons**. Our challenges in building the Experimentation Platform were both technical and cultural. The technical challenges revolved around building a highly scalable system capable of dealing with some of the most visited sites in the world (e.g., the MSN home page). However, those are engineering challenges and there are enough books on building highly scalable systems. It is the cultural challenge, namely getting groups to see experimentation as part of the development lifecycle, which was (and is) hard, with interesting lessons worth sharing. Our hope is that the lessons can help others foster similar cultural changes in their organizations.
2. **Successful experiments.** We ran controlled experiments on a wide variety of sites. Real-world examples of experiments open people's eyes to the potential and the return-on-investment. In this paper we share several interesting examples that show the power of controlled experiments to improve sites, establish best practices, and resolve debates with data rather than deferring to the HIghest-Paid-Person's Opinion (HiPPO) or to the loudest voice.
3. **Interesting statistics.** We share some sobering statistics about the percentage of ideas that pass all the internal evaluations, get implemented, and fail to improve the metrics they were designed to improve.

Our mission at the Experimentation Platform team is to accelerate software innovation through trustworthy experimentation. Steve Jobs said that "We're here to put a dent in the universe. Otherwise why else even be here?" We are less ambitious and have made a small dent in Microsoft's universe, but enough that we would like to share the learnings. There is undoubtedly a long way to go, and we are far from where we wish Microsoft would be, but three years into the project is a good time to step back and summarize the benefits.

In Section 2, we briefly review the concept of controlled experiments. In Section 3, we describe the progress of experimentation at Microsoft over the last three years. In Section 4, we look at successful applications of experiments that help motivate the rest of the paper.

In Section 5, we review the ROI and some humbling statistics about the success and failure of ideas. Section 6 reviews the cultural challenges we faced and how we dealt with them. We conclude with a summary. Lessons and challenges are shared throughout the paper.

2. Controlled Experiments

It's hard to argue that Tiger Woods is pretty darn good at what he does. But even he is not perfect. Imagine if he were allowed to hit four balls each time and then choose the shot that worked the best. Scary good.
-- Michael Egan, Sr. Director, Content Solutions, Yahoo (Egan, 2007)

In the simplest controlled experiment, often referred to as an A/B test, users are randomly exposed to one of two variants: Control (A), or Treatment (B) as shown in Figure 1: High-level flow for an A/B test (Kohavi, Longbotham, Sommerfield, & Henne, 2009; Box, Hunter, & Hunter, 2005; Holland & Cochran, 2005; Eisenberg & Quarto-vonTivadar, 2008). The key here is "random." Users cannot be distributed "any old which way" (Weiss, 1997); no factor can influence the decision.

Based on observations collected, an Overall Evaluation Criterion (OEC) is derived for each variant (Roy, 2001). The OEC is sometimes referred to as a Key Performance Indicator (KPI) or a metric. In statistics this is often called the Response or Dependent Variable.

If the experiment was designed and executed properly, the only thing consistently different between the two variants is the change between the Control and Treatment, so any statistically significant differences in the OEC are the result of the specific change, establishing causality (Weiss, 1997, p. 215).

Common extensions to the simple A/B tests include multiple variants along a single axis (e.g., A/B/C/D) and multivariable tests where the users are exposed to changes along several axes, such as font color, font size, and choice of font.

For the purpose of this paper, the statistical aspects of controlled experiments, such as design of experiments, statistical tests, and implementation details are not important. We refer the reader to the paper *Controlled experiments on the web: survey and practical guide* (Kohavi, Longbotham, Sommerfield, & Henne, 2009) for more details.

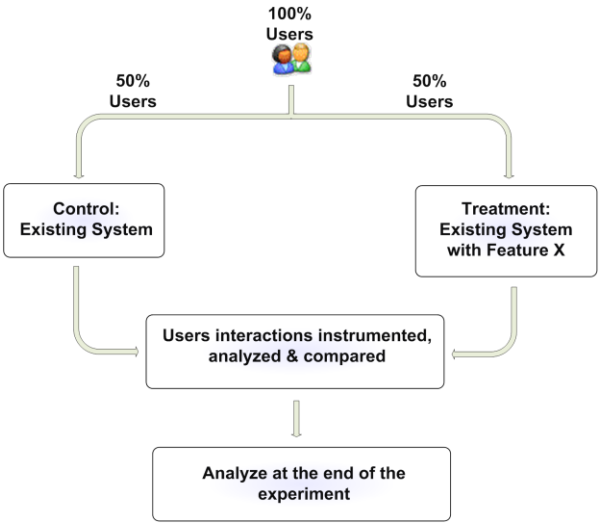


Figure 1: High-level flow for an A/B test

3. Experimentation at Microsoft

The most important and visible outcropping of the action bias in the excellent companies is their willingness to try things out, to experiment. There is absolutely no magic in the experiment… But our experience has been that most big institutions have forgotten how to test and learn. They seem to prefer analysis and debate to trying something out, and they are paralyzed by fear of failure, however small.
-- Tom Peters and Robert Waterman, In Search of Excellence (Peters & Waterman, 1982)

In 2005, when Ronny Kohavi joined Microsoft, there was little use of controlled experiments for website or service development at Microsoft outside Search and the MSN US home page. Only a few experiments ran as one-off "split tests" in Office Online and on microsoft.com. The internet Search organization had basic infrastructure called "parallel flights" to expose users to different variants. There was appreciation for the idea of exposing users to different variant, and running content experiments was even patented (Cohen, Kromann, & Reeve, 2000). However, most people did not test results for statistical significance. There was little understanding of the statistics required to assess whether differences could be due to chance. We heard that there is no need to do statistical tests because "even election surveys are done with a few thousand people" and Microsoft's online samples were in the millions. Others claimed that there was no need to use sample statistics because all the traffic was included, and hence the entire population was being tested.[1]

[1] We're not here to criticize but rather to share the state as we saw it. There were probably people who were aware of the statistical requirements, but statistics were not applied in a consistent manner, which was partly the motivation for forming the team. We also recognized that development of a single testing platform would allow sufficient concentration of effort and expertise to have a more advanced experimentation system than could be developed in many isolated locations.

In March 2006, the Experimentation Platform team (ExP) was formed as a small incubation project. By end of summer we were seven people: three developers, two program managers, a tester, and a general manager. The team's mission was dual-pronged:

1. Build a platform that is easy to integrate
2. Change the culture towards more data-driven decisions

In the first year, a proof-of-concept was done by running two simple experiments. In the second year, we focused on advocacy and education. More integrations started, yet it was a "chasm" year and only eight experiments ultimately ran successfully. In the third year, adoption of ExP, the Experimentation Platform, grew significantly. The search organization has evolved their parallel flight infrastructure to use statistical techniques and is executing a very large number of experiments independent of the Experimentation Platform on search pages, but using the same statistical evaluations.

Figure 2 shows that increasing rate of experiments: 2 experiments in fiscal year 2007, 8 experiments in fiscal year 2008, 44 experiments in fiscal year 2009.

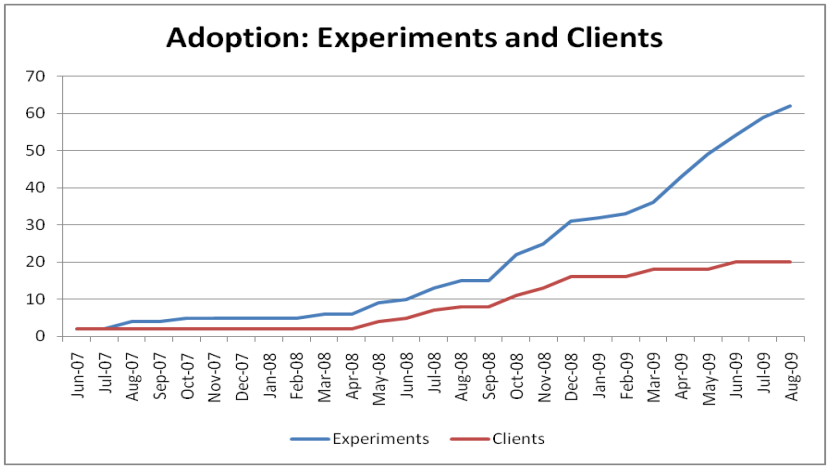


Figure 2 Adoption of ExP Services by Microsoft online properties

Microsoft properties that have run experiments include

1. HealthVault/Solutions
2. Live Mesh
3. MSCOM Netherlands
4. MSCOM Visual Studios
5. MSCOM Home Page
6. MSN Autos DE
7. MSN Entertainment
8. MSN EVS pre-roll
9. MSN HomePage Brazil
10. MSN HomePage UK
11. MSN HomePage US
12. MSN Money US
13. MSN Real Estate US
14. Office Online
15. Support.microsoft.com (PQO)
16. USBMO
17. USCLP Dynamics
18. Windows Genuine Advantage
19. Windows Marketplace
20. Xbox

Testimonials from ExP adopters show that groups are seeing the value. The purpose of sharing the following testimonials isn't self-promotion, but rather to share actual responses showing that cultural changes are happening and ExP partners are finding it highly beneficial to run controlled experiments. Getting to this point required a lot of work and many lessons that we will share in the following sections. Below are some testimonials.

- I'm thankful every day for the work we've done together. The results of the experiment were in some respect counter intuitive. They completely changed our feature prioritization. It dispelled long held assumptions about video advertising. Very, very useful.
- The Experimentation Platform is essential for the future success of all Microsoft online properties… Using ExP has been a tremendous boon for the MSN Global Homepages team, and we've only just begun to scratch the surface of what that team has to offer.
- For too long in the UK, we have been implementing changes on homepage based on opinion, gut feeling or perceived belief. It was clear that this was no way to run a successful business…Now we can release modifications to the page based purely on statistical data
- The Experimentation Platform (ExP) is one of the most impressive and important applications of the scientific method to business. We are partnering with the ExP...and are planning to make their system a core element of our mission

4. Applications of Controlled Experiments at Microsoft

Passion is inversely proportional to the amount of real information available
-- "Benford's Law of Controversy", Gregory Benford, 1980.

One of the best ways to convince others to adopt an idea is to show examples that provided value to others, and carry over to their domain. In the early days, publicly available examples were hard to find. In this section we share recent Microsoft examples.

4.1 Which Widget?

The MSN Real Estate site (http://realestate.msn.com) wanted to test different designs for their "Find a home" widget. Visitors to this widget were sent to Microsoft partner sites from which MSN Real estate earns a referral fee. Six different designs, including the incumbent (i.e. the Control), were tested, as shown in Figure 3.

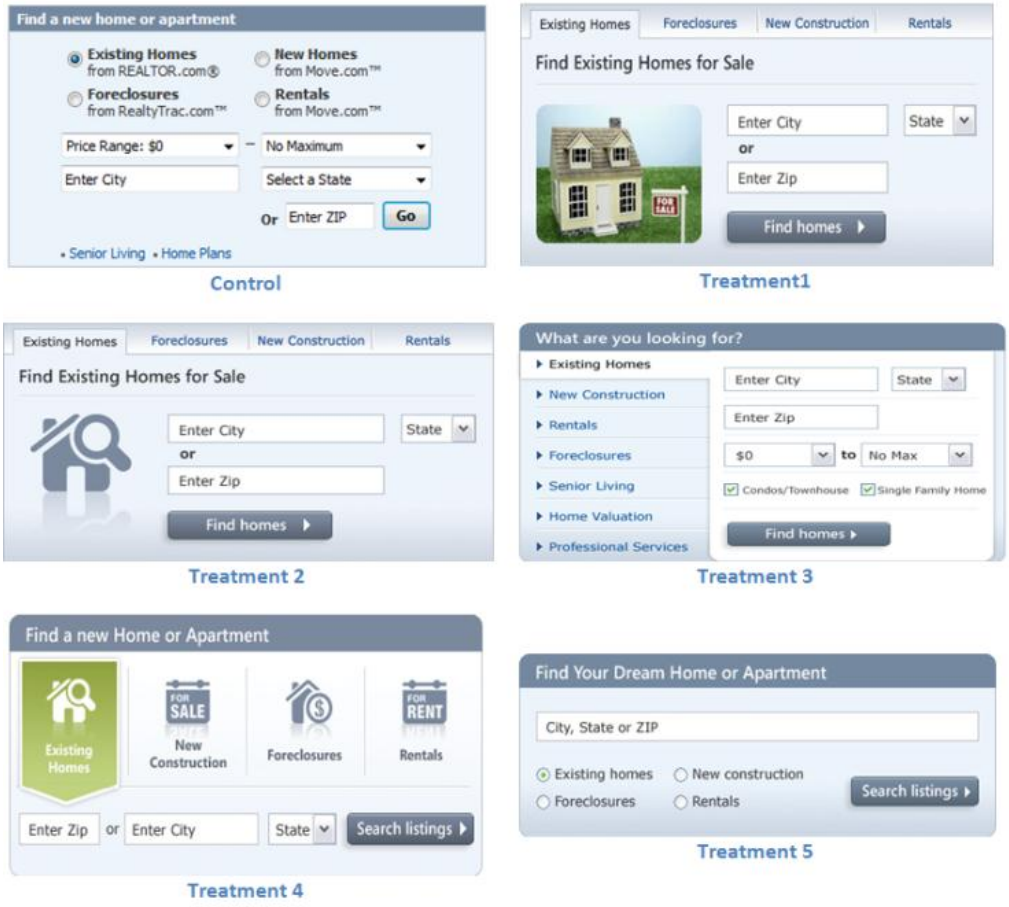


Figure 3: Widgets tested for MSN Real Estate

A "contest" was run by ZAAZ, the company that built the creative designs, prior to running an experiment, with each person guessing which variant will win. Only three out of 21 people guessed the winner. All three said, among other things, that they picked Treatment 5 because it was simpler. One person said it looked like a search experience.

The winner, Treatment 5, increased revenues from referrals by almost 10% (due to increased clickthrough).

4.2 Open in Place or in a Tab?

When a visitor comes to the MSN UK home page and they are recognized as having a Hotmail account, a small Hotmail convenience module is displayed. Prior to the experiment, if they clicked on any link in the module, Hotmail would open in the same tab/window as the MSN home page, replacing it. The MSN team wanted to test if having Hotmail open in a new tab/window would increase visitor engagement on the MSN because visitors will reengage with the MSN home page if it was still present when they finished reading e-mail.

The experiment included one million visitors who visited the MSN UK home page, shown in Figure 4, and clicked on the Hotmail module over a 16 day period. For those visitors the number of clicks per user on the MSN UK homepage increased 8.9%. This change resulted in significant increase in user engagement and was implemented in the UK and US shortly after the experiment was completed.

One European site manager wrote: "This report came along at a really good time and was VERY useful. I argued this point to my team and they all turned me down. Funny, now they have all changed their minds."

Figure 4: Hotmail Module highlighted in red box

4.3 Pre-Roll or Post-Roll Ads?

Most of us have an aversion to ads, especially if they require us to take action to remove them or if they cause us to wait for our content to load. We ran a test with MSN Entertainment and Video Services (http://video.msn.com) where the Control had an ad that ran prior to the first video and the Treatment post-rolled the ad, after the content. The primary business question the site owners had was "Would the loyalty of users increase enough in the Treatment to make up for the loss of revenue from not showing the ad up front?" We used the first two weeks to identify a cohort of users that was then tracked over the next six weeks. The OEC was the return rate of users during this six week period. We found that the return rate increased just over 2% in the Treatment, not enough to make up for the loss of ad impressions, which dropped more than 50%.

MSN EVS has a parameter, which is the minimum time between ads. We were able to show that users are not sensitive to this time and decreasing it from 180 seconds to 90 seconds would improve annual revenues significantly. The changed was deployed in the US and being deployed in other countries.

4.4 MSN Home Page Ads

A critical question that many site owners face is how many ads to place. In the short-term, increasing the real-estate given to ads can increase revenue, but what will it do to the user experience, especially if these are non-targeted ads? The tradeoff between increased revenue and the degradation of the end-user experience is a tough one to assess, and that's exactly the question that the MSN home page team at Microsoft faced.

The MSN home page is built out of modules. The Shopping module is shown on the right side of the page above the fold. The proposal was to add three offers right below it, as shown in Figure 5, which meant that these offers would show up below the fold for most users. The Display Ads marketing team estimated they could generate tens of thousands of dollars per day from these additional offers.

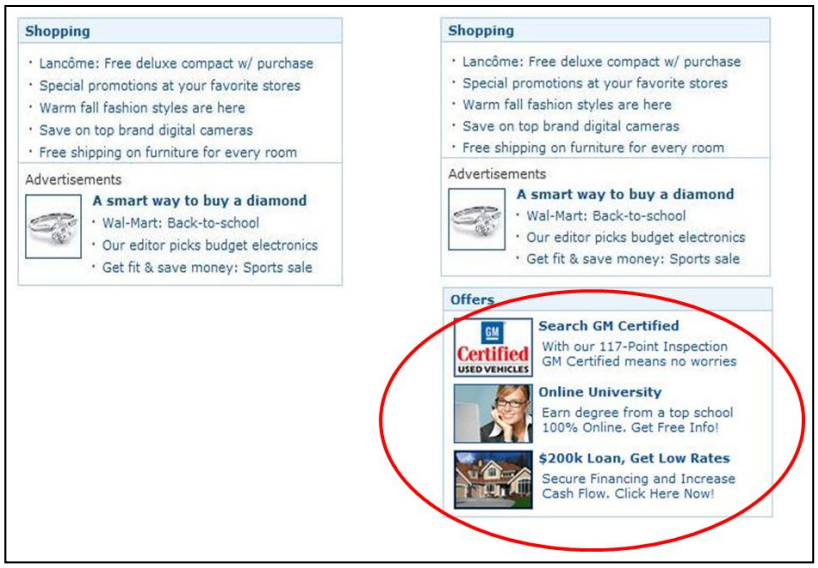


Figure 5: MSN Home Page Proposal.
Left: Control, Right: proposed Treatment

The interesting challenge here is how to compare the ad revenue with the "user experience." We refer to this problem as the OEC, or the Overall Evaluation Criterion. In this case, we decided to see if page views and clicks decreased, and assign a monetary value to each. (No statistically significant change was seen in visit frequency for this experiment.) Page views of the MSN home page have an assigned value based on ads; clicks to destinations from the MSN home page were estimated in two ways:

1. Monetary value that the destination property assigned to a click from the MSN home page. These destination properties are other sites in the MSN network. Such a click generates a visit to an MSN property (e.g., MSN Autos or MSN Money), which results in multiple page views.
2. The cost paid to search engines for a click that brings a user to an MSN property but not via the MSN home page (Search Engine Marketing). If the home page is driving less traffic to the properties, what is the cost of regenerating the "lost" traffic?

As expected, the number from #2 (SEM) was higher, as additional value beyond direct monetization is assigned to a click that may represent a new user, but the numbers were close enough to get agreement on the monetization value to use.

A controlled experiment was run on 5% of the MSN US home page users for 12 days. Clickthrough rate decreased by 0.35% (relative change), and the result was statistically significant. Page views per user-day decreased 0.35%, again a result that was highly statistically significant.

Translating the lost clicks to their monetary value, it was higher than the expected ad revenue. The idea of placing more ads was appropriately stopped.

4.5 Personalize Support?

The support site for Microsoft (http://support.microsoft.com) has a section near the top of the page that has answers to the most common issues. The support team wanted to test whether making those answers more specific to the user would be beneficial. In the Control variant, users saw the top issues across all segments. In the Treatment, users saw answers specific to their particular browser and operating system. The OEC was the click-through rate (CTR) on the links to the section being tested. The CTR for the treatment was over 50% higher the Control, proving the value of simple personalization. This experiment ran as a proof of concept with manually generated issue lists. The support team now plans to add this functionality to the core system.

4.6 MSN Homepage US Search Header Experiment

The search header experiment tested replacing the magnifying glass with three actionable words: Search, Go, Explore. Below is the Search variant.

Usability people have taught us that buttons should have an actionable word. Steve Krug's Don't Make Me Think makes this very clear. The folks at FutureNow picked up a prior experiment we did and suggested that we change the magnifying glass to "Search" 9 months ago.

The data is supports this well: the results showed that all three treatments with actionable words were better than the magnifying glass on all key metrics. Use of "Search" statistically significantly increased searches by 1.23%.

4.7 Search Branding Experiment

This experiment was carried out to test a change to the Search header at the top of the MSN Homepage prior to the official launch of Bing, so the Bing name could not be used. This experiment informed the final design of the Search header for the Bing launch (compare the Control in Figure 6 to the Treatment in Figure 7).

The Treatment increased the percent of users who clicked on the Search box, the number of searches as well as the number of clicks to the whole page.

Figure 6 Control for Search branding experiment

Figure 7 Treatment for Search branding experiment

4.8 More Information

One of our goals at ExP is to share results widely and enable the "institutional memory" about what worked and what did not so groups at Microsoft can stand on the shoulder of others instead of stepping on their toes. We now send a weekly e-mail with a summary of an interesting experiment to our internal discussion group. ExP holds a monthly one-day seminar called Planning and Analysis of Online Experiments, which is a great way to learn more about the theory and practical applications of experimentation.

5. The ROI for Experimentation

The fascinating thing about intuition is that a fair percentage of the time it's fabulously, gloriously, achingly wrong

-- *John Quarto-vonTivadar, FutureNow*

There are psychological experiments where subjects are shown a series of lights with two colors: green and red, and are asked to guess the next color each time. The red light appears 75% of the time and the green light appears 25% of the time. One could choose two reasonable strategies: (i) guess the guess the color that appears more frequently, a route favored by rats and other nonhuman animals, or (ii) match the guesses to the proportions observed, a route preferred by humans. If the colors are shown randomly, the first strategy leads to a 75% success rate, but one concedes a 25% loss; the second strategy, preferred by humans who attempt to find the hidden patterns (where none exist), leads to only a 62.5% success rate. Humans therefore commonly lose to a rat (Mlodinow, 2008).

Section 5.1 below shows that despite our best efforts and pruning of ideas, most fail to show value when evaluated in controlled experiments. The literature is filled with reports that success rates of ideas in the software industry, when scientifically evaluated through controlled experiments, are below 50%. Our experience at Microsoft is no different: only about 1/3 of ideas improve the metrics they were designed to improve. Of course there is some bias in that experiments are run when groups are less sure about an idea, but this bias may be smaller than most people think; at Amazon, for example, it is a common practice to evaluate every new feature, yet the success rate is below 50%.

Some people believe that teams will discover the good and bad ideas after they launch, even if they do not run a controlled experiment. This is valid only for changes that are either extremely good or extremely bad. For most ideas, the change in key metrics will be too small to detect over time. Section 5.2 shows that attempt to cut corners and run sequential experiments are ill-advised, as it is very likely that external effects will dwarf the effect one attempts to detect. Finally, if a team is not testing the idea, but rather launching it, backing things out is expensive. When Office Online changed their rating system from yes/no to 5 stars, they lost over 80% of responses. It took eight months to detect, analyze, and replace that version! If a metric drops by 3%, the chances that anyone will discover it and start a project to back out a feature they proudly launched is miniscule.

How do you value an experiment then? (We are looking at the value of an experiment to Microsoft, not attempting to assign specific percentages to the team running the experiment and the experimentation platform itself. At the end of the day, we are one Microsoft and over time the cost of running experiments will go down as more and more self-service capabilities are built and more integration is done to enable experiments through the underlying systems.)

The value of an experiment is really the delta between the perceived value of the treatment prior to running the experiment, and the value as determined in the controlled experiment. Clearly the team that develops an idea thinks it is a useful idea, so there are four possibilities.

1. The idea is as good as the team thought it would be. In this case, the experiment adds little value. As shown below, this case is uncommon.
2. The idea is thought to be good, but the experiment shows that it hurts the metrics it was designed to improve. Stopping the launch saves the company money and avoids hurting the user experience. As humbling as it may be, this represents about one third of experiments.
3. The idea is thought to be good, but it does not change the metrics it was designed to improve significantly (flat result). In this case we recommend stopping the launch. There is always a cost to additional deployments, and the new code may not be QA'ed as well. In fact, this is one of the reasons to launch early prototypes. For example, if you reduce the QA matrix by only certifying the feature for Internet Explorer and run the experiment only for IE users, you could learn much sooner that the feature is not useful for the majority of your users, enabling a significant time saving because, as the saying goes, it's the last 20% of the effort (in this case supporting several other browsers) that takes 80% of the time. Our experience indicates that about 1/3 of experiments are flat.
4. The idea is thought to be good, but through experiments, it turns out to be a breakthrough idea, improving key metrics more than expected. The org can then focus on launching it faster, improving it, and developing more ideas around it. At Amazon, an intern project called Behavior-Based Search turned out to be so beneficial due to early experiments that resources were quickly diverted into the idea, resulting in revenue improvements worth hundreds of millions of dollars. This case is also rare, but that's basically a given or else it would not be a "breakthrough."

A team that simply launches 10 ideas without measuring their impact may have about 1/3 be good, 1/3 flat, and 1/3 negative (matching our current estimates on the ExP team). The overall value of the 10 ideas will therefore be fairly small. On the other hand, if the team experiments with the 10 ideas and picks the successful three or four, aborting the rest, the benefits will be significant. Even though running an experiment has costs, the ability to abort bad features early and fail fast can save significant time and allow teams to focus on the truly useful features.

5.1 Most Ideas Fail to Show Value

It is humbling to see how bad experts are at estimating the value of features (us included). Every feature built by a software team is built because *someone* believes it will have value, yet many of the benefits fail to materialize. Avinash Kaushik, author of *Web Analytics: An Hour a Day*, wrote in his Experimentation and Testing primer (Kaushik, 2006) that "80% of the time you/we are wrong about what a customer wants." In Do It Wrong Quickly (Moran, Do It Wrong Quickly: How the Web Changes the Old Marketing Rules , 2007, p. 240), the author writes 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!" (Moran, Multivariate Testing in Action, 2008).

We in the software business are not unique. QualPro, a consulting company specializing in offline multi-variable controlled experiments, tested 150,000 business improvement ideas over 22 years and reported that 75 percent of important business decisions and business improvement ideas either have no impact on performance or actually hurt performance (Holland & Cochran, 2005). In the 1950s, medical researchers started to run controlled experiments: "a randomized controlled trial called for physicians to acknowledge how little they really knew, not only about the treatment but about disease" (Marks, 2000, p. 156). In *Bad Medicine: Doctors Doing Harm Since Hippocrates*, David Wootton wrote that "For 2,400 years patients have believed that doctors were doing them good; for 2,300 years they were wrong." (Wooton, 2007). 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 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 (Kohavi, Bloodletting: Why Controlled Experiments are Important , 2008). In an old classic, Scientific Advertising (Hopkins, 1923, p. 23), the author writes that "[In selling goods by mail] false theories melt away like snowflakes in the sun... One quickly loses his conceit by learning how often his judgment errs--often nine times in ten."

When we first shared some of the above statistics at Microsoft, many people dismissed them. Now that we have run many experiments, we can report that Microsoft is no different. Evaluating well-designed and executed experiments that were designed to improve a key metric, **only about one-third were successful at improving the key metric!**

There are several important lessons here

1. Avoid the temptation to try and build optimal features through extensive planning without early testing of ideas. As Steve Kurg write: "The key is to start testing early (it's really never too early) and test often, at each phase of Web development" (Krug, 2005).
2. Experiment often. Because under objective measures most ideas fail to improve the key metrics they were designed to improve, it is important to increase the rate of experimentation and lower the cost to run experiments. Mike Moran phrased this lesson as follows: "You have to kiss a lot of frogs to find one prince. So how can you find your prince faster? By finding more frogs and kissing them faster and faster" (Moran, 2007).
3. A failure of an experiment is not a mistake: learn from it. Badly executed experiments are mistakes (Thomke S. H., 2003), but knowing that an idea fails provides value can save a lot of time. It is well known that finding an error in requirements is 10 to 100 times cheaper than changing features in a finished product (Boehm, 1981)(McConnell, 2004). Use experimentation with software prototypes to verify requirements in the least costly phase of the software development lifecycle. Think of how much effort can be saved by building an inexpensive prototype and discovering that you do not want to build the production feature at all! Such insights are common in organizations that experiment. The ability to fail fast and try multiple ideas is the main benefit of a customer-driven organization that experiments frequently. We suggest that development teams launch prototype features regularly, and extend them, making them more robust, and then fully deploy them only if they prove themselves useful. This is a challenging proposition for organizations whose development culture has been to "do it right the first time".
4. Try radical ideas and controversial ideas. In (Kohavi, Longbotham, Sommerfield, & Henne, 2009), we described the development of Behavior-Based Search at Amazon, a highly controversial idea. Early experiments coded up by an intern showed the surprisingly strong value of the feature, which ultimately helped improve Amazon's revenue by 3%, translating into hundreds of millions of dollars in incremental sales. Greg Linden at Amazon created a prototype to show personalized recommendations based on items in the shopping cart (Linden, Early Amazon: Shopping cart recommendations, 2006). Linden notes that "a marketing senior vice-president was dead set against it," claiming it will distract people from checking out. Greg was "forbidden to work on this any further." Nonetheless, Greg ran a controlled experiment and the rest is history: the feature was highly beneficial. Multiple sites have

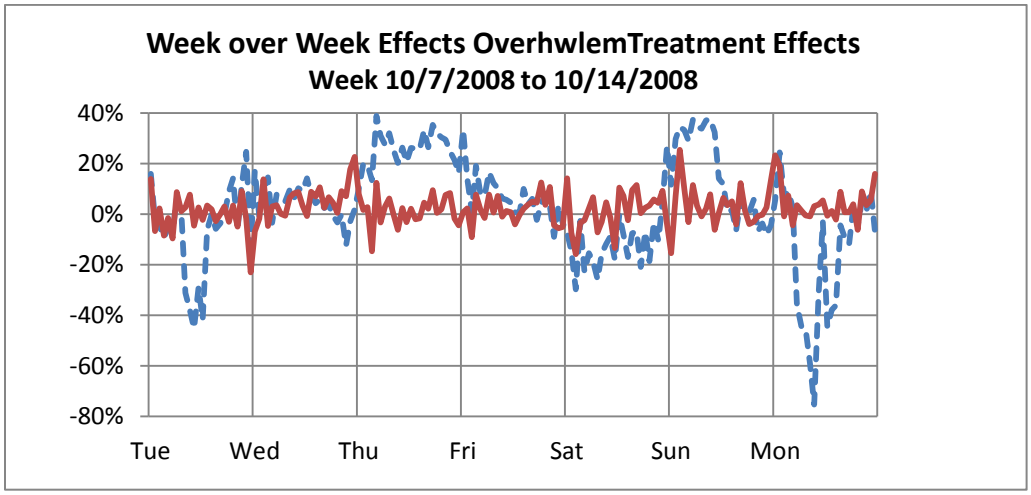
copied cart recommendations. Sir Ken Robinson made this point eloquently when he said: "If you're not prepared to be wrong, you will not come up with anything original" (Robinson, 2006).

5. Experiment to find out how a feature degrades the user experience to make business decisions. Experiments are not limited to testing hypothetically positive effects. The example of adding ads to the MSN home page in Section 4.4 is one that was thought to degrade the user experience, but a financial decision needed to be made to tradeoff revenue vs. some degradation (the surprise was the amount of degradation). Many online applications and services ask users to jump through hoops as part of processes such as authentication and signup (signup funnels are notoriously prone to this problem.) Product managers, enraptured by the promise of rich user profile data, may not appreciate the negative effect of these sorts of features. Experiments can be used to find out whether the tradeoffs of additional revenue or data are acceptable.

5.2 Running Sequential (Uncontrolled) Experiments

Teams which face challenges integrating controlled experiments into their applications frequently ask if they can realize the benefits of experimentation by taking an easier approach. For example, they may propose measuring Control during week one and Treatment during week two. The following analysis, based on data taken from a two week MSN Real Estate controlled experiment, shows the risks in a sequential approach. The metric of interest is click-through rate (clicks/page views) and the chart below plots

1. Blue/dashed: Treatment from Week 2 minus Control from Week 1 (week over week)
2. Red/solid: Treatment from Week 2 minus Control from Week 2 (same week)



As one can visually see, the blue line has higher variability than the effect we are trying to detect, and this is very common. The following table shows the magnitude of the problem.

Effect	% Change
Week 1: Treatment Week 1 minus Control Week 1	1.93%
Week 2: Treatment Week 2 minus Control Week 2	2.09%
Both Weeks: Treatment minus Control	2.06%
Week over Week: Treatment Week 2 minus Control Week 1	11.38%

Treatment performed 1.93% better than Control in Week 1 and 2.09% better than Control in Week 2 (absolute deltas). Over both weeks Treatment performed 2.06% better than control. However, uncontrolled differences between Week 1 and Week 2 result in an apparent advantage of 11.38% for Treatment when the effect is calculated week-over-week. Clearly, the magnitude of the effect is due to external factors; it would be poor practice indeed to use this analysis for business decision making.

6. Cultural Challenges

There were three ways to get fired at Harrah's:
steal, harass women, or institute a program or policy without first running an experiment

-- *Gary Loveman, quoted in Hard Facts* (Pfeffer & Sutton, 2006, p. 15)

Microsoft clearly knows how to build and ship classical "shrink-wrapped" or "client" software. There have been over 120 million Office licenses sold since the launch of Office 2007 to July 2008 (Elop, 2008). Office releases are well planned and executed over three years. But in the evolving world of the web and services, there is a different way of "shipping" software. Mark Lucovsky described it well (Lucovsky, 2005):

> *When an Amazon engineer fixes a minor defect, makes something faster or better, makes an API more functional and complete, how do they "ship" that software to me? What is the lag time between the engineer completing the work, and the software reaching its intended customers? A good friend of mine investigated a performance problem one morning, he saw an obvious defect and fixed it. His code was trivial, it was tested during the day, and rolled out that evening. By the next morning millions of users had benefited from his work*

Google Apps' product manager Rishi Chandra said in an interview (Boulton, 2009)

> *In terms of the innovation curve that we have, we release features every two weeks. That is fundamentally what is going to be Google's differentiation here. We can continue to react very quickly to product trends and update the products themselves*

Websites and services can iterate faster because shipping is much easier. In addition, getting implicit feedback from users through online controlled experiments is something that could not be done easily with shrink-wrapped products, but can easily be done in online settings. It is the combination of the two that can make a big difference in the development culture. Instead of doing careful planning and execution, one can try many things and evaluate their value with real customers in near-real-time.

Linsky and Heifetz in *Leadership on the Line* (Linsky & Heifetz, 2002) describe *Adaptive Challenges* as those that are not amenable to standard operating procedures and where the technical know-how and procedures are not sufficient to address the challenge. We faced several non-technical challenges that are mostly cultural. It is said that the only population that likes change consists of wet babies. We share the things we did that we believe were useful to change the culture toward an experimentation culture.

6.1 Education and Awareness

People have different notions of what "experiment" means, and the word "controlled" in front just doesn't help to ground it. In 2005, no Microsoft groups that we are aware of ran proper controlled experiments with statistical tests.

Figure 8: Example of a Poster: Experiment or Die!

In the few groups that ran "flights," as they were called, traffic was split into two or more variants, observations were collected and aggregated, but no tests were done for statistical significance, nor were any power calculations done to determine how large a sample was needed and how long experiments should run. This led to overfitting the noise in some cases.

One of our first challenges was education: getting people to realize that what they have been doing was insufficient. Upton Sinclair wrote that "It is difficult to get a man to understand something when his salary depends upon his not understanding it." People have found it hard to accept that many of their analyses, based on raw counts but no statistics, have been very "noisy," to put it mildly.

We started teaching a monthly one-day class on statistics and design of experiments. Initially, we couldn't fill the class (of about 20), but after a few rounds interest grew. To date more than 500 people at Microsoft have attended our class, which now commonly has a waiting list.

The challenge is ongoing, of course; we still find people who test ideas by comparing counts from analytical reporting tools without controlling for many factors and without running statistical tests.

We wrote the KDD paper *Practical Guide to Controlled Experiments on the Web* (Kohavi, Henne, & Sommerfield, Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO, 2007) in our first year to help give the team credibility as "experts" in the field. The paper is now part of the class reading for several classes at Stanford University (CS147, CS376), USCD (CSE 291), and at the University of Washington (CSEP 510). It is getting referenced by dozens of articles and some recent book, such as King (Website Optimization: Speed, Search Engine & Conversion Rate Secrets , 2008).

We put posters across the Microsoft campus with examples of A/B tests or with quotations. One of the more controversial and successful ones was "Experiment or Die!," shown in Figure 8, with a fossil and a quotation from Hal Varian at Google.

We ate our own dog food and A/B tested our posters by creating two designs for each promotion. Each design was tagged with a unique URL offering more information about our platform, services and training classes. We compared page views for each URL to determine the effectiveness of each design.

One of our most successful awareness campaigns featured a HiPPO stress toy imprinted with our URL. HiPPO stands for the Highest Paid Person's Opinion (Kohavi, Henne, & Sommerfield, Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO, 2007). We gave away thousands of HiPPOs at the annual Microsoft employee company meetings, in our training classes, introductory talks, and through a HiPPO FAQ web site.[2] The campaign went viral, spawning word of mouth awareness and even a small fan club in Microsoft India.

We created an internal Microsoft e-mail distribution list for those interested in experimentation. There are now over 1,300 people on the exp-int list.

In June 2007, MSW Inside Track published a story titled Microsoft Faces Challenge to Experiment or Die, which was then re-published again later that year with following editor's note: "This article originally ran on June 26, 2007, and was one of the most popular Inside Track stories of the year."

In early 2008, Mary-Jo Foley published an interview with Ronny Kohavi (Foley, 2008). Traffic to our external site http://exp-platform.com grew from an average of 45 visitors/day to an average over 500/day for the week following the publication. A recent presentation at Seattle Tech Startups was blogged by Greg Linden and resulted in hundreds of new visitors.

6.2 Perceived Loss of Power

Linsky and Heifetz wrote that "People do not resist change, per se. People resist loss" (Linsky & Heifetz, 2002). Some people certainly viewed experimentation as a risk to their power and/or prestige. Some believed it threatened their job as decision makers. After all, program managers at Microsoft select the next set of features to develop. Proposing several alternatives and admitting you don't know which is best is hard. Likewise, editors and designers get paid to create a great design. In some cases an objective evaluation of ideas may fail and hurt their image and professional standing.

It is easier to declare success when the feature launches and not *if* it is liked by customers. We have heard statements such as "we know what to do. It's in our DNA," and "why don't we just do the right thing?" This was, and still is, a significant challenge, despite the humbling statistics about the poor success rate of ideas when evaluated objectively (see Section 5).

What we found was that a great way to convince people that we are not good at predicting the outcomes of experiment is to challenge them. We created a survey with eight A/B tests, and offered a nice polo shirt for anyone who could correctly guess 6 out of 8 (the options were: A is statistically significantly better, B is statistically significantly better, or there's no statistically significant difference between them). With over 200 responses, we didn't have to hand out a single shirt! 6 out of 200 had 5 answers correct; the average was 2.3 correct answers. Humbling! At the 2008 CIKM conference (Pasca & Shanahan, 2008), Kohavi gave an invited talk on controlled experiments and challenged the audience to predict the outcome of three actual A/B tests that ran. Out of about 150 people in the audience who stood up to the challenge, only 1 correctly guessed the outcome of two challenge questions[3]. Note that with three options to each question, this is much worse than random (150/9 = 16 people).

6.3 Reward Systems

Lee et al. (The Mixed Effects of Inconsistency on Experimentation in Organizations, 2004) write about the mixed effects of inconsistency on experimentation in organizations. They note that management can support experimentation and highlight it as a value (normative influence), but inconsistent reward systems that punish failure lead to aversion, especially in organizations that are under constant evaluation for perfect execution.

At Microsoft, as in many other large companies, employees are evaluated based on yearly goals and commitments. Conventional wisdom is that the best goals and commitments need to be SMART: specific, measurable, attainable, realistic and timely[4]. Most goals in software development organizations at Microsoft are around "shipping" products, not about their impact on customers or key metrics. In most projects, the classical triangular tradeoff exits between features, time, and quality. Some teams, such as Microsoft Office, traditionally

[2] See http://exp-platform.com/whatsahippo.aspx

[3] A video recording of the presentation with the live quiz is available at http://videolectures.net/cikm08_kohavi_pgtce/

[4] Guy Kawasaki in Reality Check (2008, p. 94) suggests that goals be "rathole resistant," to avoid short-term target that dead-ends. We agree, and we have emphasized the importance of setting OECs for long-term customer lifetime-value.

focused on time and quality and cut features; others focused on features and quality and delayed the release schedule. Either way, features are commonly defined by their perceived value and are prioritized by program managers. Controlled experiments and the humbling results we shared bring to question whether a-priori prioritization is as good as most people believe it is. One possible change to goal definitions is to avoid tying them to products and features, but rather tie them to key metrics, and empower the development organizations to regularly test their ideas using controlled experiments. The feature development pace will undoubtedly slow down, but more corrections will be made on the way, ultimately leading to a better customer experience in shorter time. Amazon does not announce major redesigns—the site improves continuously and is considered one of the most innovative sites.

It is hard for us to judge whether we are making any change in people's goals; cultural changes take time and it is unlikely that we have made a dent in many people's yearly performance goals. This is an ongoing challenge worth highlighting.

6.4 Website Performance: Timing is Critical

Small delays can have dramatic impact on user behavior and therefore be very expensive, yet that is often not appreciated. Greg Linden (Linden, Make Data Useful, 2006, p. 15) wrote that experiments at Amazon showed a 1% sales decrease for an additional 100msec. In Google reveals speed as a secret to its success, they write

> It turns out that a delay as short of the blink of an eye — about 400 milliseconds — could turn users off. Last year, Mayer said, Google experimented by injected a 400 millisecond delay into its delivery of search results. Searches per user started dropping. After six weeks, searches per user had fallen nearly one percent. That seemingly small figure represented several hundred million dollars a year in potential ad revenue, Mayer noted.

Experiments at Microsoft Live Search (Kohavi, Emetrics 2007 Practical Guide to Controlled Experiments on the Web, 2007, p. 12) showed that when the search results page was slowed down by one second, queries per user declined by 1% and ad clicks per user declined by 1.5%; when the search results page was slowed down by two seconds, these numbers more than doubled to 2.5% and 4.4%. Additional research was presented by Microsoft's Eric Schurman in Velocity 2009. Many sites do not have sufficient focus on performance.

6.5 Robots Impact Results

Web sites are accessed not only by human users but also by robots such as search engine crawlers, email harvesters and botnets. The traffic generated by robots is not representative of the human population (e.g., excessive clicks and page views in patterns that differ from human patterns) and can cause misleading results.

For example, in an early experiment on the MSN home page, where a small change was done to only one module, we found that the click-through rate on several areas of the page were statistically significantly different. Since the change was small and localized to one area of the page, we were surprised to see significant differences in unrelated areas. Upon deeper investigation, we found that the differences were caused by robots that accept cookies and execute JavaScript. Executing code in JavaScript is one of the most common characteristics that separate humans from robots, and some web analytic vendors even claim that page tagging using JavaScript is so robust that no additional robot detection should be done. Yet in this case these robots were executing JavaScript "onclick" events, which fire on the MSN portal when users click a link on a web page, at extremely high rates of about 100 per minute for durations of 2.5 hours.

Robots implemented by automating browsers such as Internet Explorer or Firefox support all of the functionality of those browsers including cookies and JavaScript. Furthermore, when such a robot runs from a machine also used by a human, both the robot and human will typically share the same cookies. If the user identity is stored in a cookie (very common), then the user appears to be schizophrenic, acting like a human at certain times and like a robot at others.

Note that the example above is not an isolated case. Most experiments we run are impacted by robots. Furthermore, robots impact not only the results of experiments but any analysis based on web traffic. Identifying and removing robots is difficult but is critical in order to produce valid results (Tan, et al., 2002; Kohavi, et al., 2004; Bomhardt, et al., 2005; Bacher, et al., 2005; Wikipedia: Internet bot, 2008; Wikipedia: Botnet, 2008).

We have seen many groups, both internal and external to Microsoft, make incorrect simplifying assumptions about robots (e.g., robots don't execute JavaScript) or even ignore robots completely. The cultural challenge here is for groups to dedicate the necessary time and effort to remove robots during web analysis. Since robot detection is a difficult problem, the best approach is to leverage the work of another group that already does this (e.g., Search or ExP).

6.6 Failed A/A Tests

Groups that start to experiment may have existing bugs or have a bug in their integration, which bias experiments. To keep the quality bar high, we execute A/A tests, which basically split users but show them exactly the same page. If the integration is correct, then there should be no difference between the two variants on key metrics like pages per user or users per variant. Here are some notable examples of A/A test failures that were caught.

6.6.1 Microsoft.com Caching

After running an A/A test on the home page, we noticed an unexpected discrepancy between the number of visitors that were being shown the Treatment versus those shown the Control. The problem was tracked to the page caching key, which did not include a bit for whether the user had been previously seen (with an MC1 cookie) or not. As it turned out, the instrumentation code to track users was only emitted for users with a cookie, so depending on whether the first user to the home page for a given web server had a cookie or not, the instrumentation was on or off for all future users until the process was recycled.

6.6.2 Support.Microsoft.com (PQO) Cookie Loss

After running an A/A test on the support site, we saw more users in the control than in the treatment. A bug had the effect that some new users (without a cookie) who were internally assigned a cookie and assigned to variant were not properly issued the cookie. That meant that on their next request, they were re-randomized. The bug existed prior to the A/A test, which meant that PQO's reports were showing incorrect numbers.

6.6.3 Unbalanced Redirection

When running an experiment that redirects users (e.g., issues a 302), one needs to ensure that it is not setup to disadvantage one of the variants. For example in a simple A/B test where A is the control and B is the new treatment, if one is presenting control immediately without a redirect and redirecting for the treatment, there will be a delay of a few hundred milliseconds for the treatment.

This delay impacts the end user and increases the chance of abandonment. During an experiment we run with Office Online, a redirection was done only for the treatment. The A/A test failed with Control having more page views. When we reran the experiment with the same redirection applied to both variants, the audit passed. It is worth noting that most third party experimentation system, including Google Website Optimizer in whole-page (A/B testing mode), have this problem.

6.7 Incorrect Reasons Not to Experiment

Controlled experiments are a tool that has its limitations, which we discussed in *Controlled experiments on the web: survey and practical guide* (Kohavi, Longbotham, Sommerfield, & Henne, 2009). A recent article by Davenport (How to Design Smart Business Experiments, 2009) points out that controlled experiments are best suited for strategy execution, not strategy formulation; they are not suited for assessing a major change in business models, a large merger or acquisition (e.g., you can't run a randomized experiments on whether Microsoft should acquire Yahoo!). We agree, of course. However, we have also heard many incorrect reasons why not to experiment and would like to address them.

1. Claim: Experimentation leads to incremental innovations.
   While it is true that one can limit experiments to trivial UI changes like choosing colors, there is no reason experiments can't be used for radical changes and non-UI changes. Amazon makes heavy use of experimentation and its page design has evolved significantly—its first home page did not even have a search box. Multiple industry-leading innovations came from experimenting with prototypes that showed significant value and were reprioritized quickly once their value was apparent. Two such examples were described in Section 5.1 (item 4). One lesson that we learned is that many of our initial examples were indeed highlighting a big difference achieved through a small UI change, something that may have solidified the thinking that experimentation is best used for small incremental changes. Now we emphasize more sophisticated examples, such as whether to show ads (Section 4.3) and backend changes (Section 4.5).
2. Claim: Team X is optimizing for something that is not measurable.
   Here we need to differentiate between not measurable and non-economical to measure. We believe that the former is a bad way to run a business. If you can't articulate what you're optimizing for, how can the organization determine if you are doing a good job? If you are not improving a measurable metric, perhaps the other direction is also true: no measurable change will be observable without you in the organization!
   The other interpretation is more reasonable: it may be non-economical to measure the change. While this is valid at times, we would like to point to Amazon as an example of a company that did decide to measure something hard: the value of TV ads. After a 15-month-long test of TV advertising in two markets, it determined that TV ads were not a good investment and stopped them (Bezos, 2005). Is your organization avoiding experiments whose answer they would rather not know?
3. Claim: It's expensive to run experiments.
   Holland (Breakthrough Business Results With MVT: A Fast, Cost-Free, "Secret Weapon" for Boosting Sales, Cutting Expenses, and Improving Any Business Process , 2005) wrote that based on 150,000 business improvement ideas over 22 years, "there is no correlation between what people in the organization think will work and what actually does work… The lack of correlation between what people think will work and what does work has nothing to do with the level of the people in the organization who make these judgments. The experts are no better than the front-line workers or senior executives in determining which ideas will improve results." While we think Holland's sample is biased because his consulting company, QualPro, is brought in to help evaluate more controversial ideas, we do believe that people and organizations are overly confident of their ideas, and the poor success rate described in Section 5 strongly supports that. While it is expensive to experiment, it is more expensive to continue

developing and supporting features that are not improving the metrics they were supposed to improve, or hurting them, and at Microsoft, these two cases account for 66% of experiments.

The flip side is to reduce costs and develop infrastructure to lower the cost of experimentation, and that's why we embarked on building the Experimentation Platform.

7. Summary

It doesn't matter how beautiful your theory is, it doesn't matter how smart you are. If it doesn't agree with experiment[s], it's wrong.

-- *Richard Feynman*

Experimentation lies at the heart of every company's ability to innovate (Thomke S. , 2001; Thomke S. H., 2003). Running physical experiments is relatively expensive, so companies have had to be parsimonious with the number of experiments. The electric light bulb required more than 1,000 complex experiments. In modern times, with the magic of software, experimentation is much cheaper, and the ability to test innovative ideas unprecedented.

Changing the culture at a large company like Microsoft, with about 90,000 employees is not easy. As more software is written in the form of services and web sites, the value of running controlled experiments and getting direct feedback in near-real-time rises. In the last three years, experimentation at Microsoft grew significantly in usage, but we are only at the early stages. We presented successful applications of experimentation, the many challenges we faced and how we dealt with them, and many lessons. The humbling results we shared in Section 5 bring to question whether a-priori prioritization is as good as most people believe it is. We hope this will help readers initiate similar changes in their respective organizations so that data-driven decision making will be the norm, especially in software development for online web sites and services.

Acknowledgments

We would like to thank members of the Experimentation Platform team at Microsoft. Special thanks to David Treadwell, Ray Ozzie, and Eric Rudder; without their support the experimentation platform would not have existed.

REFERENCES

Agarwal, D., Chen, B.-C., Elango, P., Motgi, N., Park, S.-T., Ramakrishnan, R., et al. (2008). Online Models for Content Optimization. *Neural Information Processing Systems (NIPS)* .

Bezos, J. (2005, January). The Zen of Jeff Bezos. (C. Anderson, Ed.) *Wired Magazine* (13.01).

Boehm, B. W. (1981). *Software Engineering Economics.* Prentice-Hall.

Boulton, C. (2009, Sept 1). *Google Apps Product Manager Discusses the Collaboration War with Microsoft.* Retrieved from eweek.com: http://www.eweek.com/c/a/Messaging-and-Collaboration/Google-Apps-Product-Manager-Discusses-The-Collaboration-War-With-Microsoft-416905/

Box, G. E., Hunter, J. S., & Hunter, W. G. (2005). *Statistics for Experimenters: Design, Innovation, and Discovery* (2nd ed.). John Wiley & Sons, Inc.

Cohen, J. S., Kromann, P. K., & Reeve, T. S. (2000). *Patent No. Patent 7,343,390.*

Davenport, T. H. (2009). How to Design Smart Business Experiments. *Harvard Business Review* (February).

Egan, M. (2007, February 10). *Improving Ad Quality, Part II*. Retrieved from Yahoo! Search Marketing Blog: http://www.ysmblog.com/blog/2007/02/10/improving-ad-quality-part-ii/

Eisenberg, B., & Quarto-vonTivadar, J. (2008). *Always Be Testing: The Complete Guide to Google Website Optimizer.* Sybex .

Elop, S. (2008, July 24). *Financial Analyst Meeting 2008*. Retrieved from Microsoft Investor Relations: http://www.microsoft.com/msft/speech/FY08/ElopFAM2008.mspx

Foley, M.-J. (2008, April 16). *Microsoft looks to make product planning more science than art*. Retrieved from ZDNet: All About Microsoft: http://blogs.zdnet.com/microsoft/?p=1342

Holland, C. W., & Cochran, D. (2005). *Breakthrough Business Results With MVT: A Fast, Cost-Free, "Secret Weapon" for Boosting Sales, Cutting Expenses, and Improving Any Business Process* . Wiley.

Hopkins, C. (1923). *Scientific Advertising.* New York City: Crown Publishers Inc.

Kaushik, A. (2006, May 22). *Experimentation and Testing: A Primer.* Retrieved 2008, from Occam's Razor: http://www.kaushik.net/avinash/2006/05/experimentation-and-testing-a-primer.html

Kawasaki, G. (2008). *Reality Check: The Irreverent Guide to Outsmarting, Outmanaging, and Outmarketing Your Competition.* Portfolio Hardcover .

King, A. (2008). *Website Optimization: Speed, Search Engine & Conversion Rate Secrets* . O'Reilly Media, Inc.

Kohavi, R. (2008, May 19). *Bloodletting: Why Controlled Experiments are Important* . Retrieved from http://exp-platform.com/bloodletting.aspx

Kohavi, R. (2007, October 16). *Emetrics 2007 Practical Guide to Controlled Experiments on the Web.* Retrieved from http://exp-platform.com/Documents/2007-10EmetricsExperimenation.pdf

Kohavi, R., & Round, M. (2004). Front Line Internet Analytics at Amazon.com. (J. Sterne, Ed.) Santa Barbara, CA.

Kohavi, R., Henne, R. M., & Sommerfield, D. (2007, August). Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO. (R. C. Pavel Berkhin, Ed.) pp. 959-967.

Kohavi, R., Longbotham, R., Sommerfield, D., & Henne, R. M. (2009, February). Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery , 18* (1), pp. 140-181.

Krug, S. (2005). *Don't Make Me Think: A Common Sense Approach to Web Usability* (2nd ed.). New Riders Press.

Lee, F., Edmondson, A. C., Thomke, S., & Worline, M. (2004). The Mixed Effects of Inconsistency on Experimentation in Organizations. *Organization Science , 15* (3), pp. 310-326.

Linden, G. (2006, April 25). *Early Amazon: Shopping cart recommendations*. Retrieved from Geeking with Greg: http://glinden.blogspot.com/2006/04/early-amazon-shopping-cart.html

Linden, G. (2006, Dec). *Make Data Useful.* Retrieved from http://home.blarg.net/~glinden/StanfordDataMining.2006-11-29.ppt

Linsky, M., & Heifetz, R. (2002). *Leadership on the Line: Staying Alive Through the Dangers of Leading.* Harvard Business School Press.

Lucovsky, M. (2005, February 12). *Shipping Software* . Retrieved from Markl's Thoughts : http://mark-lucovsky.blogspot.com/2005/02/shipping-software.html

Marks, H. M. (2000). *The Progress of Experiment: Science and Therapeutic Reform in the United States, 1900-1990.* Cambridge University Press.

McConnell, S. C. (2004). *Code Complete* (2nd Edition ed.). Microsoft Press.

Mlodinow, L. (2008). *The Dunkard's Talk: How Randomness Rules Our Lives* . Pantheon.

Moran, M. (2007). *Do It Wrong Quickly: How the Web Changes the Old Marketing Rules* . IBM Press.

Moran, M. (2008, December 23). *Multivariate Testing in Action*. Retrieved from Biznology Blog by Mike Moran: http://www.mikemoran.com/biznology/archives/2008/12/multivariate_testing_in_action.html

Pasca, M., & Shanahan, J. G. (2008, Oct 29). *Industry Event*. Retrieved from ACM 17th Conference on Information and Knowledge Management : http://cikm2008.org/industry_event.php#Kohavi

Peters, T. J., & Waterman, R. H. (1982). *In Search of Excellence: Lessons from America's Best-Run Companies.* HarperCollins Publishers.

Pfeffer, J., & Sutton, R. I. (2006). *Hard Facts, Dangerous Half-Truths, and Total Nonsense: Profiting from Evidence-Based Management.* Harvard Business School Press.

Ray Ozzie. (2005, 10 28). *Ozzie memo: 'Internet services disruption'*. Retrieved from http://news.zdnet.com/2100-3513_22-145534.html

Robinson, K. (2006, Feb). Do schools kill creativity? *TED: Ideas Worth Spreading* .

Roy, R. K. (2001). *Design of Experiments using the Taguchi Approach : 16 Steps to Product and Process Improvement.* John Wiley & Sons, Inc.

Thomke, S. (2001, Feb). Enlightened Experimentation: The New Imperative for Innovation.

Thomke, S. H. (2003). Experimentation Matters: Unlocking the Potential of New Technologies for Innovation.

Weiss, C. H. (1997). *Evaluation: Methods for Studying Programs and Policies* (2nd ed.). Prentice Hall.

Wikepedia. (2008). *Multi-armed Bandit*. Retrieved from Wikipedia: http://en.wikipedia.org/wiki/Multi-armed_bandit

Wooton, D. (2007). *Bad Medicine: Doctors Doing Harm Since Hippocrates* . Oxford University Press.