Laws of Productivity

8 productivity experiments you don't need to repeat.

This deck covers 8 common workplace topics and offers suggestion on how you should approach them. I've been collecting data on and off for years and thought it would be nice to have it all in one presentation friendly spot. Feel free to use pieces of this presentation as the need arises.

So you want to maximize your productivity?

- Fixed release schedule.
- Fixed resources.
- Way too much work to finish.
- Failure = Demotion, Job loss, or Pariah status

'Maximizing productivity' is another way of asking to the team to squeeze water from a stone. The stone consists of the hard resource and time constraints the team operates under. Naturally, the cost of failure to pull off a productivity miracle is couched in terms that most people wish to avoid.

What is Productivity?

- + Work accomplished
- - Work required to fix defects
- - Work required to fix bad design decisions

It is possible for productivity to be negative when workers end up doing more harm than good.

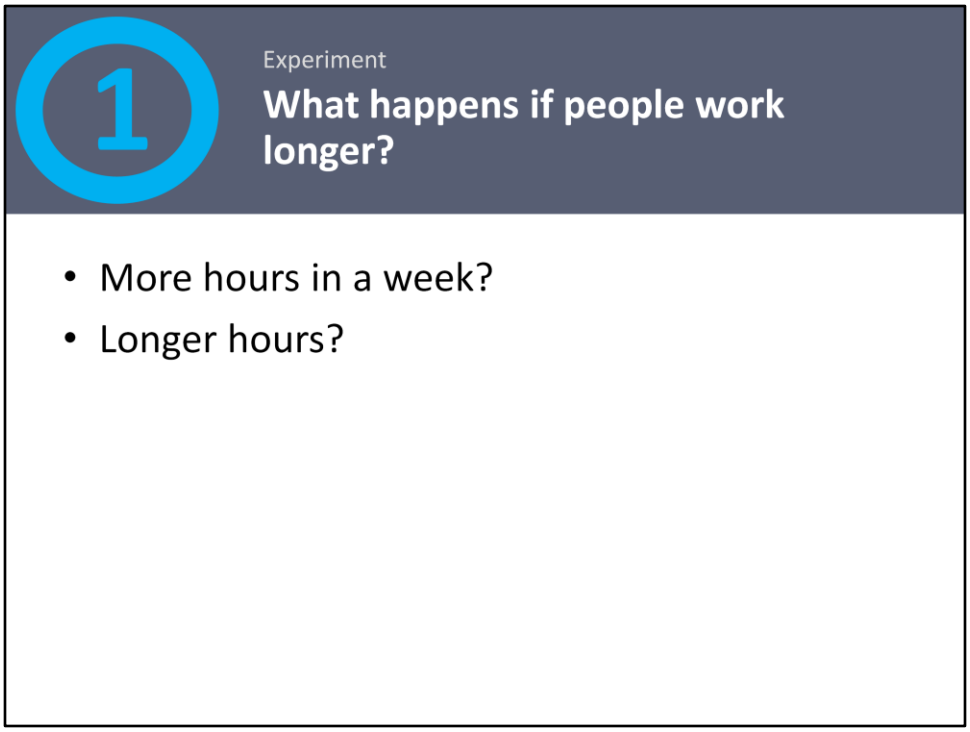
People commonly measure ‘what was accomplished’, but often this is a poor measure of productivity. It is possible to check in code and design decisions that must be later fixed or removed at great cost. If you only measure work accomplished, you could generate great ‘productivity’ numbers but never ship a working product. The real measure of productivity is valued working code in customer hands.

Why reinvent the wheel?

Businesses have been running productivity experiments since the early 1900's.

- Ford ran 12-years of productivity experiments.
- These have been repeated over the past century in hundreds of industries.
- Even the game industry.

Most managers plow through the act of management with few guidelines other than gut or habit on what works best. Luckily there is a large body of well researched material out there on what works. Not all of it agrees. However, there are some obviously broken concepts such as overtime that should be put to rest. And there are some practical ideas like small team sizes that offer a big boost for relatively little effort.



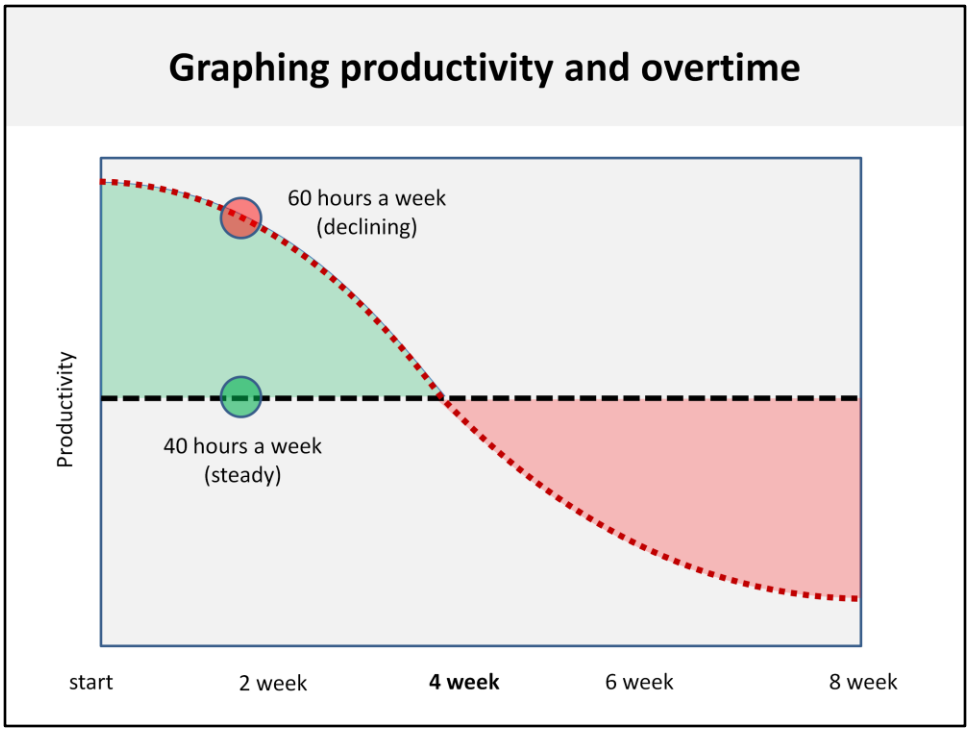
Let’s start with the most common productivity boosting strategy, overtime.

Results 1

Working more than 40 hours a week leads to decreased productivity

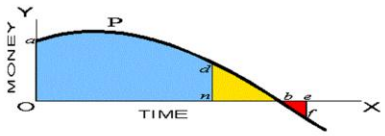
- <40 hours and people weren't working enough.
- >60 hour work week gives a small productivity boost.
- The boost lasts 3 to 4 weeks and then turns negative.

Ford chewed on this problem for 12 years and run dozens of experiments. As a result of Ford's experiments, he and his fellow industrialists lobbied Congress to pass 40 hour a week labor laws. Not because he was nice. Because he wanted to make the most money possible. We like to think of a 40 hour work week as a 'liberal policy' when in fact it was hard headed capitalism at it's finest.



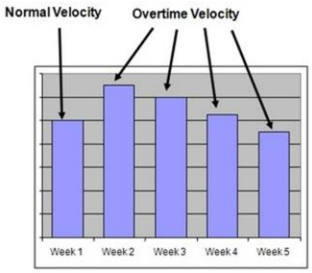
I love this graph. It shows the complete idiocy of crunching for long periods of time.

Work has changed, people have not.



Factory work - 1909

Sidney J. Chapman's *Hours of Labour* (1909)



Game development - 2005

http://www.agilegamedevelopment.com/2008/06/scrum-overtime.html

Same Curve.

You'll have to imagine the curve here. Chapman is using money produced as a measure of productivity. Highmoon is using ideal hours as a measure of productivity. Both graphs show that productivity goes negative after an initial boost in productivity.

Lessons

- Work 40-hour weeks with time for rest and family.
- Never work 2 months of 60-hour crunch.
  - It accomplishes less, despite the initial boost.

Experiment

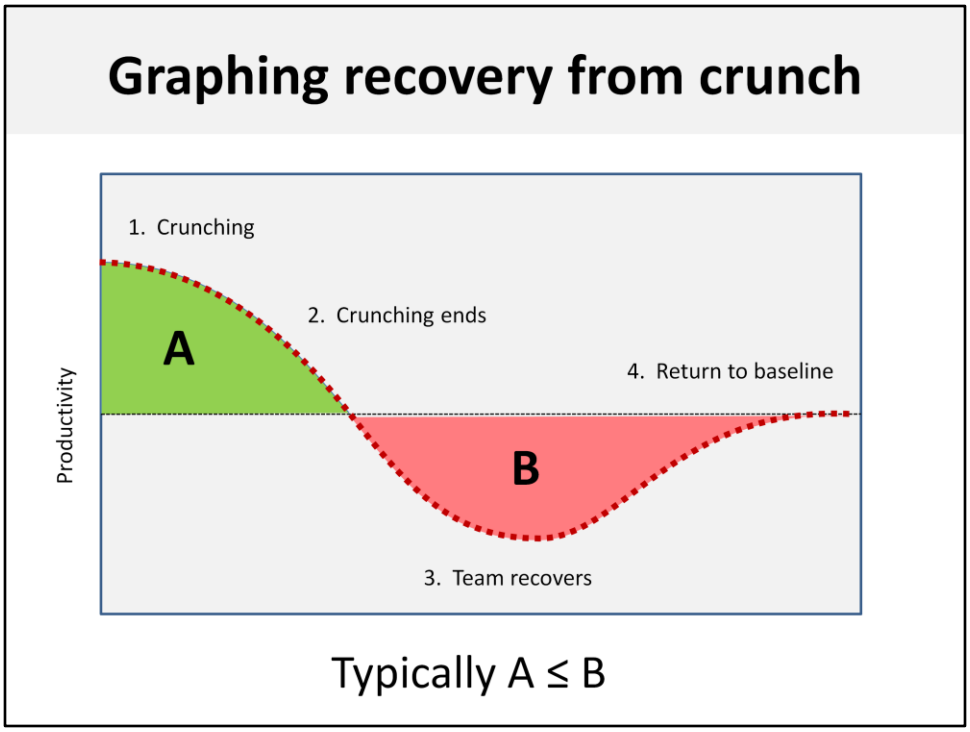
What happens if we work harder in bursts?

- Can we take advantage of the burst that comes from working overtime?
- What happens if we crunch for a week and then work ‘only’ 40 hours for another week?
- Are there other patterns of scheduling work that might be more efficient?

Results 2

There is always a cost to crunch

- Anything over 40 hours results in a recovery period, not matter how you split it up.
- 35 to 40 hour weeks can be divided in a variety of ways, such as four 10-hour days on and three days off.
- These ‘compressed work weeks’ can reduce absenteeism and, in some cases, increase productivity 10 to 70%



When you crunch, you pay.

Lessons

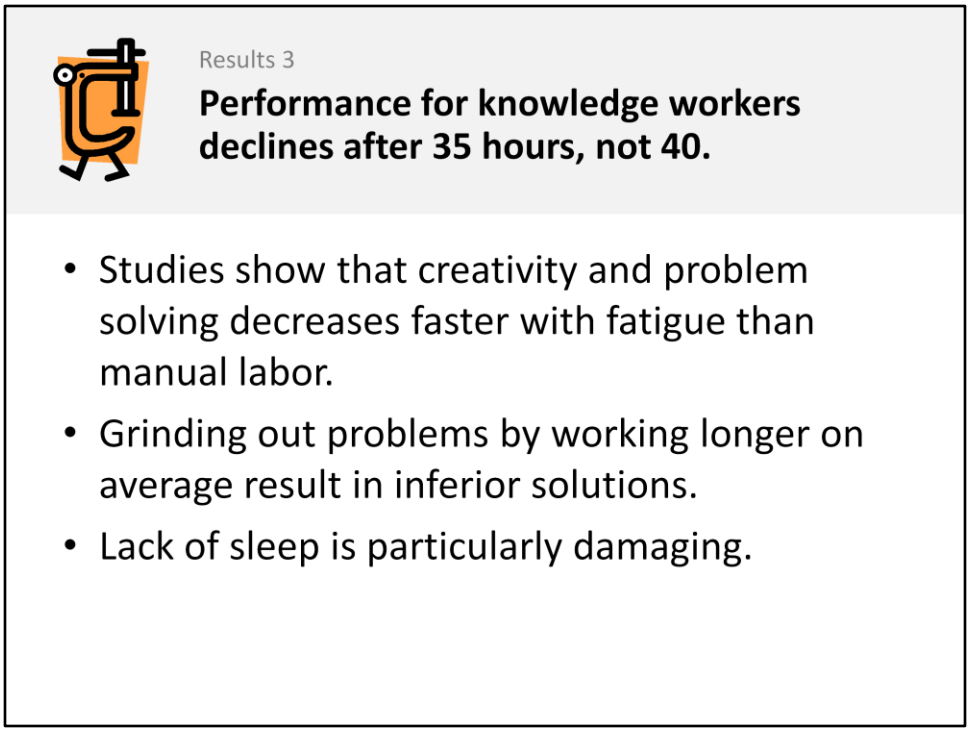
- Short spurts <3 weeks can raise productivity temporarily.
- Team can use overtime tactically to meet nearby deadlines.
- Plan for an equivalent reduction in productivity immediately afterwards.
- Consider 4-day work weeks as a flextime option.

From http://www.shrm.org/hreducation/Creating_Flex_Work_PPT_Final.ppt
Research has shown no negative effect on worker productivity when employees participate in flexible work options. In fact, there is growing evidence that flexible work has a positive effect on productivity.
A meta-analysis of 31 flexible work studies found that flexible schedules increased employee productivity and lowered absenteeism (Baltes, et.al., 1999).
In the National Study of the Changing Workforce, a study by the Families and Work Institute, 39 percent of employees with high availability of flexible work arrangements reported "high levels of loyalty and the willingness to work harder than required to help their employers succeed" (Bond, Thompson, Galinsky, & Prottas, 2002, p. 34).
According to the National Work Life Measurement Project, approximately one-third of managers said their work group was more productive because it included employees who used flexible work arrangements (Fried, Litchfield, & Pruchno, 2003, p. 36).

Experiment

Repeat experiments on knowledge workers, not factory workers

- Games require creativity and problem-solving, not manual labor.
- Do the same rules apply?

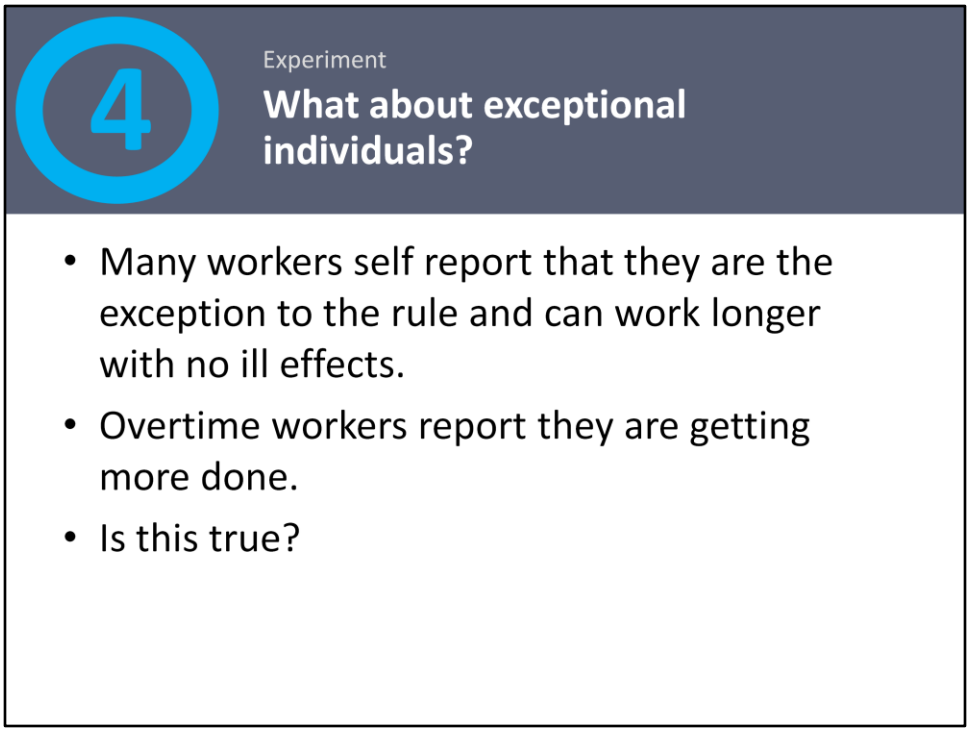


There is some evidence that knowledge workers should only be working 35 hour work weeks. Past this, they start becoming tired and making dumb decisions. This sort of talk makes business owners trained by Ford's experiments from 90 years ago very uncomfortable. Our well trained 'gut' assumes that less than 40 hour workweeks means that people are slacking.

One of my favorite stories here is that business owner who realized that for certain jobs she could hire two part-time workers for less than the cost of two full time workers and still get the same amount of work done. There was less absenteeism and workers stayed on task. All at a much lower cost of course.

Lessons

- Overtime kills creativity.
- If you are stuck on a problem, go home or take a break.
- Get 8 hours of sleep. Sleep substantially improves your problem solving abilities.



This is the most common objection I hear when I present this data. People love to think that they are exceptional superhumans. Particularly young single men with something to prove.

Results 4

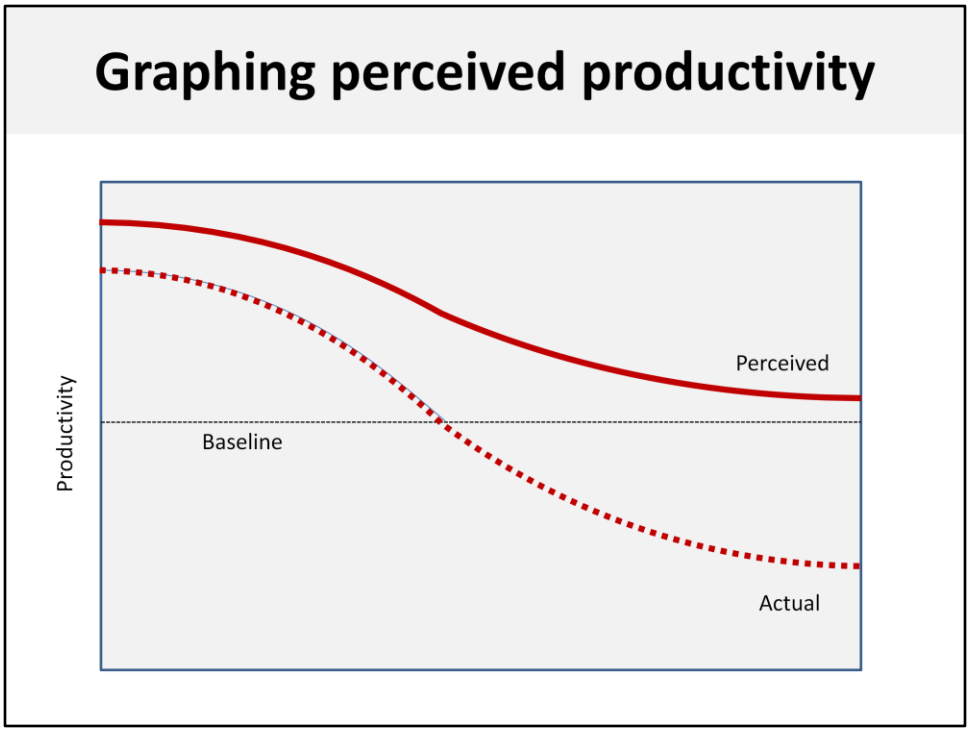
Teams on overtime feel like they are doing more, but actually accomplish less

Perform objective measurements:

- Team A on overtime and Team B without.
- Team A *feels* like they are doing much more than Team B.
- Yet, Team B produces the better product.

This assumes that the teams are roughly equal. This scenario has been played many times in the tech industry. One of the more fascinating side effects is the perception of both teams with management. Since Team A has more 'butts in seats', they are often given more resources, promoted more and generally considered to be better workers. Unfortunately, due to the burden that overtime places on workers, this is not a self fullfilling prophecy. It is quite common that managers put all their eggs in the basket that ends up self destructing.

The solution is better metrics so that managers can gain a more objective understanding of how projects are doing.



In a 60 hour crunch people have a vague sense that they are doing worse, but never think that they should stop crunching. They imagine that working 40 hours a week will decrease their productivity. In fact, it will let them rest and increase their productivity.

This behavior is fascinating to observe. Zombies stumble over to their desk every morning. Temper flare. Bugs pour in. Yet to turn back would be a betrayal.

Why?

Humans ignore the systematic costs and psychological biases

- **Failure to measure**…
  - Cost of defects
  - Cost of bad design decisions
  - Missed opportunities
- **Linear extrapolation**: If workers see an initial burst of productivity from overtime, they assume they'll get the same from future effort.
- **Habit**: This is what we've always done.
- **Self reported excellence**: Behavior rewarded independent of actual results.

These cognitive biases are all exacerbated by lack of sleep. Crunch makes smart people stupid. Especially if you are 23 and have very little work experience.

Lessons

- The feeling of increased long-term productivity is **false**.
- Rest your exceptional producers to achieve even better results.
- Use customer metrics to determine actual productivity

If you have a programmer who can churn through 10 features in a week working overtime, imagine how far your project would be along if he was programming 8 features that were actually the right ones instead of 5 that needed to be ripped out and reworked?

Experiment

What is the most productive team size for developers

- Does productivity change for various team sizes?
- Which size team produces the best product?

Results

Productivity is maximized in small teams of 4-8 people.

- Productivity for small groups is shown to be 30–50% higher than groups over 10.
- Cost of communication increases dramatically for groups larger than 10.
- Smaller groups don’t have enough breadth to solve a wide array of problems well

The numbers vary slightly here. Some studies claim teams of 3 are good. Others say 9 isn’t so bad. But in general 4 to 8 seems to work well.

Lessons

- Split your projects into small cross-functional teams.
- Use ‘Scrum-of-scrums’ to link small teams together on larger projects
- Create a process for:
  - Growing new teams
  - Splitting large teams
  - Transitioning to new projects

Experiment

What is the most productive physical work environment?

- Are cubes, individual offices or team rooms most effective?
- Every *individual* has an opinion, but what is best for the *team*?

Results 6

Seat people on the same team together in a closed team room

- Studies show 100% increase in Productivity.
- Being nearby means faster communication and better problem-solving.
- Fewer external interruptions to the team (not the individual) means higher productivity.

Seating people together is probably one of the biggest productivity bangs for the buck that you can implement.

Lessons

- Seat the team in their own room. With walls.
- Give at least 50 square feet per person. Less reduces productivity.
- Create side rooms for private conversations, phone calls and meeting with external groups.
- Minimize non-team distractions.

The constraints are important here. If you cram 10 people in a cubicle and call it a ‘team space’ you aren’t going to get many of the benefits. There is a sweet spot.

Experiment

How should workers of different disciplines be organized?

- Should teams be composed of a single discipline? For example, all programmers or all artists?
- Or should teams be mixed?

Results 7

Cross-functional teams outperform siloed teams

- Produced more effective solutions in the same time.
- More likelihood of generating breakthrough solutions.
- There is some negotiation of norms up front, but this is a short-term loss.

Why?

The right people are in the room

- Fewer external dependencies mean fewer lengthy blockages.
- Team has the breadth to see the forest, not just the trees.
- Different perspectives mean lower chance of groupthink.

Interestingly enough groupthink also tends to appear when groups become larger. Since so much effort is required to maintain group cohesion, alternative ideas are less tolerated.

Lessons

Create teams where every skill needed to solve the problem at hand is in the same room.

- **Limit the charter**: “Do everything = big team”.
- **Fulltime**: Focuses team member efforts. Multitasking = 15% drop in efficiency.

Experiment

What percentage of team capacity should be officially scheduled?

- 110% to promote people to ‘stretch’?
- 100% because that is what they can do?
- 80% because slacking is good?

Results

Scheduling at 80% produces better products

- Scheduling people at 100% doesn't give space to think of creative solutions.
- Not lost time: Passionate workers keep thinking.
- The 20% goes into new idea generation and process improvements
- Producing 20 great features is usually far more profitable than producing 100 competent features.

We see a theme appear throughout product development literature. It is better to produce a product with a small, but well-chosen feature set than a product with a large ill-targeted feature set. This leads to some counter intuitive work practices. Improved productivity is less focused on 'doing more' and instead looks at ways people can 'work smarter'

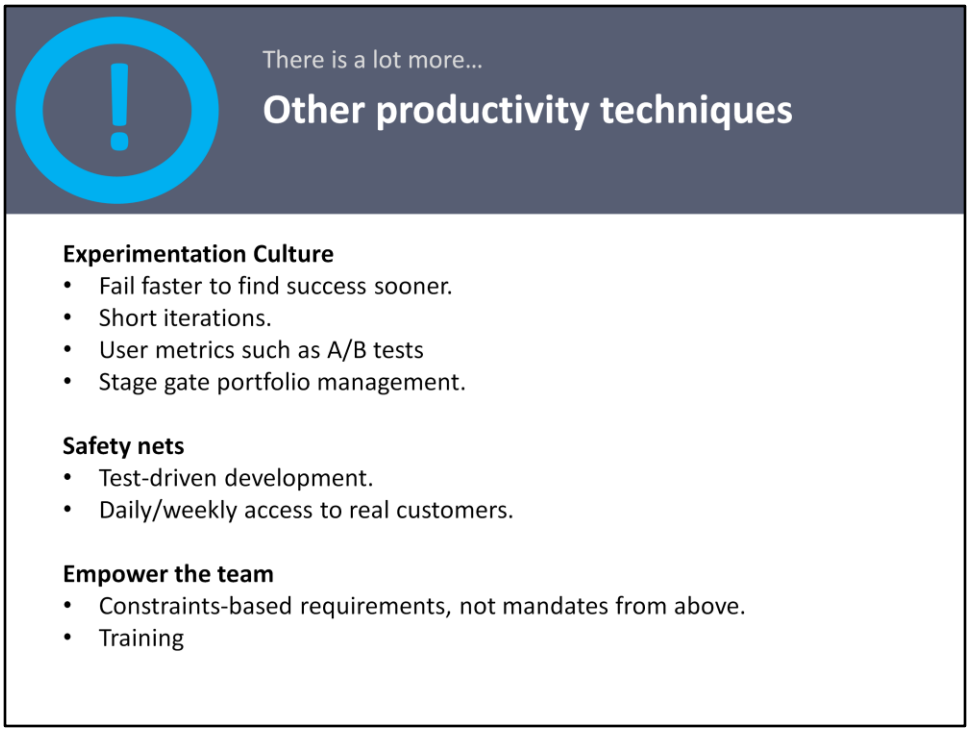
Why?

Gives time to work on important, but not urgent tasks

- Allows employees to explore many options cheaply.
- Allows people time to prototype breakthrough solutions that sound crazy on paper.
- Allows people to pursue passions.

Lessons

- Schedule 20% below possible velocity if you are running a scrum team.
- Hold periodic reviews of side projects and award interesting ideas.
- Publicize and reward side projects that make their way into production.
- Keep a public list of important things if anyone runs out of work (happens rarely).



These 8 experiments are just the start. Improving productivity is a continuous process that must always be adapted to the human beings on the teams and reality of the situation at hand.

References

Crunch in the game industry
- http://www.igda.org/articles/erobinson_crunch.php
- http://www.infoq.com/news/2008/01/crunch-mode

Best team size
- http://knowledge.wharton.upenn.edu/article.cfm?articleid=1501
- http://www.teambuildingportal.com/articles/systems-approaches/teamperformance-teamsize.php

Sleep and problem solving
- http://www.cnn.com/2004/HEALTH/01/21/sleep.creativity.ap/index.html

Sickness and Overtime correlation
- http://cat.inist.fr/?aModele=afficheN&cpsidt=15461524

Prioritization
- http://en.wikipedia.org/wiki/First_Things_First_(book)

4 day work week
- http://findarticles.com/p/articles/mi_m1093/is_1_42/ai_53697784
- http://rop.sagepub.com/cgi/reprint/28/2/166
- From http://www.shrm.org/hreducation/Creating_Flex_Work_PPT_Final.ppt

Team spaces
- "Rapid Software Development through Team Collocation" IEEE Transactions on Software Engineering, Volume 28, No. 7, July 2002