

How to be a 10X Engineer

Tips and Tricks from a Geek

About Me

- Founder of ChangeTip
- Mentor at Plug & Play and 500Startups
- 16 years in software, 4x VP of Engineering
- Have been called a 15x engineer
- Obsessed about productivity
- Software craftsman
-

Nick Sullivan - @gorillamania - nick@sullivanflock.com

Slides available at the end



WELL HERE COMES AN

AVALANCHE OF BULLSHIT

imgflip.com

What is this talk about?

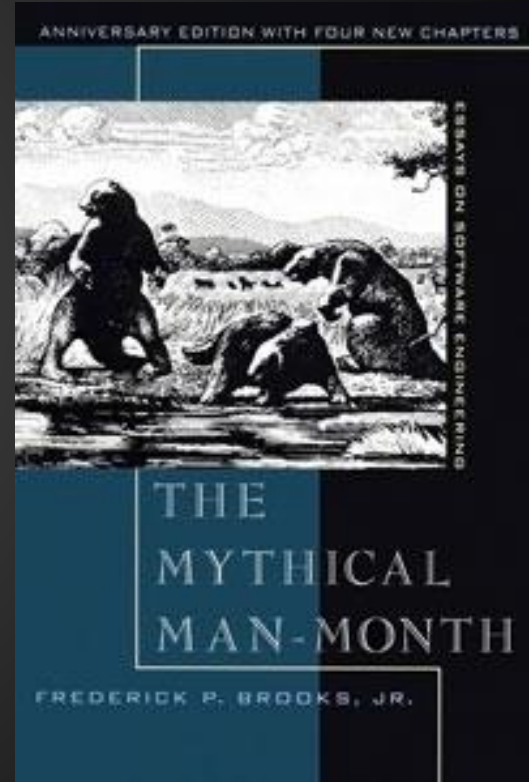
- Definition
- How-to
- Tooling
- Focus

The Software Process

What we do is hard

The Mythical Man-Month

- The Bible of software engineering management
- The process
- Software team dynamics



What is a 10x Engineer

If you get 10 engineers in a room, the best will be **100x** more effective than the worst.

This has implications on all phases of the Software Development Life Cycle.



Understanding the problem

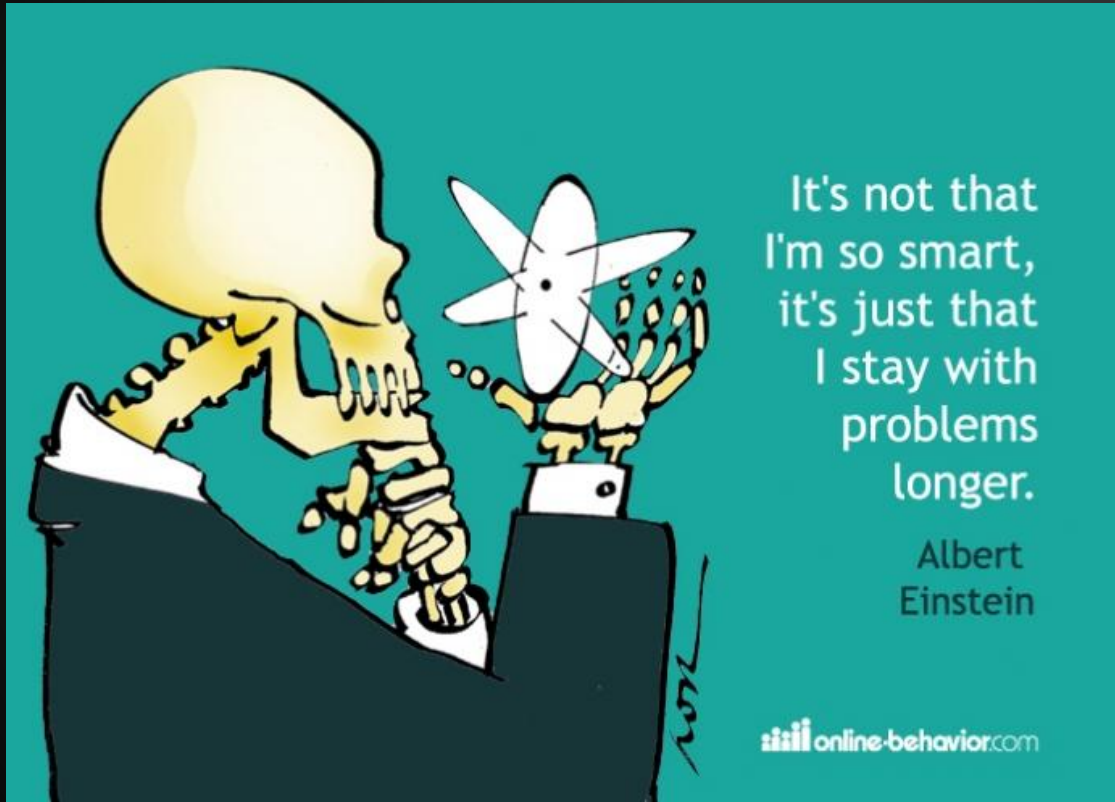
“Most people do not listen with the intent to understand; they listen with the intent to reply.”

Stephen R. Covey
(1932-2012)
InspirationBoost.com

Gathering proper requirements is a skill. Listening can be learned.

Pro Tip - README-driven development

Solving the problem



It's not that
I'm so smart,
it's just that
I stay with
problems
longer.

Albert
Einstein

Pro Tip: Architecture
Diagrams FIRST

Communication,
documentation, and
planning in one step

www.lucidcharts.com

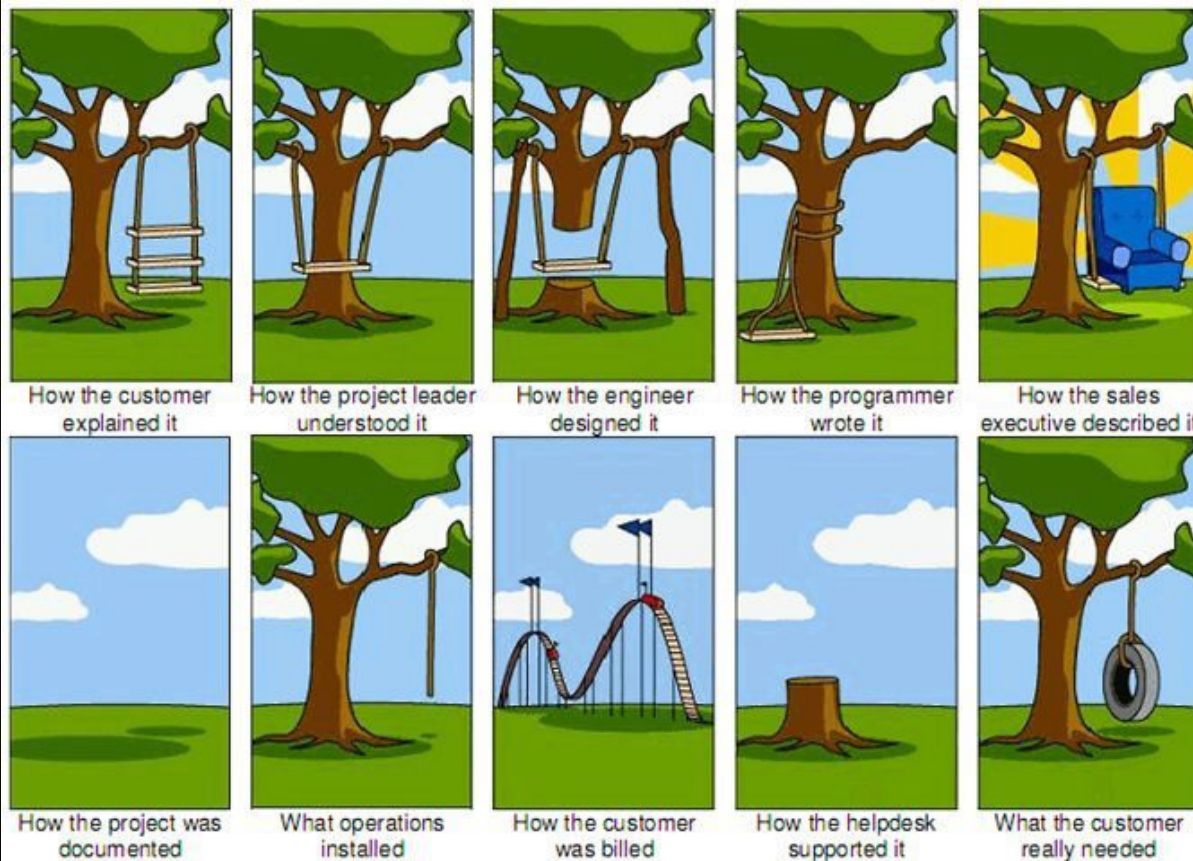
Estimating the scope

- Yes, estimating is hard - <http://qr.ae/vsLbf>
- But... A 10x engineer will be able to give good estimates instinctively
- People aren't looking for accuracy/deadlines, they are looking for easy/medium/hard
 - Pro tip - give estimates in hours, days, or weeks

Speed of development

- Proficiency
 - Familiarity with the tech stack
 - avoid new-framework-of-the-month
 - Experience (right tool for the job)
 - Tooling
- Decision Making
 - Taking the right shortcuts (especially with startups)
 - Predicting & catching rat-holes
- Debugging

Does the solution match the problem



Pro Tip - Lean Startup helps here.

TDD for product development

Simplicity (maintainability)

“If you can't explain it to a six year old, you don't understand it yourself.”

- Albert Einstein

“Any intelligent fool can make things bigger, more complex, and more violent. It takes a touch of genius — and a lot of courage to move in the opposite direction.”

- E.F. Schumacher

#1 metric for code quality is simplicity.

DON'T:

- Unnecessarily optimize
- Be clever
- Be terse

Robustness

- Does it work?
 - Tests, verification
- Does it scale?
 - Rule of thumb - 10x without changes
- Does the Ops team like deploying it?
 - Health checks, status, logging
- Is it extensible?
 - If you are continually refactoring, see Understanding the Problem

Communication

- Most conflict/problems are the result of poor communication
- Small, highly competent groups of people build software
- Each new team member is someone else to “Get on the same page”

It's not linear

“I’ve never ***** a 10, But I’ve ***** 5 2s”

- George Carlin

Especially considering communication expense

Lower than an 8 is a net negative for the team

Be a 10xer

Accelerate your team

Employee Math

- *Dividers* cause dysfunction in the company
 - Fire them immediately!
- *Subtractors* - Pull everyone down
 - New role or fire
- *Adders* - the bare minimum to contribute
- *Multipliers* - accelerate the entire team!
 - Keep them happy

10xers

- Champion for Productivity
- Champion for Culture
 - Take everyone out drinking (we bond over drink)
- Critique is cheap; craftsmanship is priceless
 - Be the one people complain to. Fix it.
- Be a talent magnet - attract others

10x character formula

4 must-have attributes

1. Intelligence
2. Passion
3. Integrity
4. Communication (can be taught)

Focus your interview process on *all* 4.

Create a culture of celebration



<http://nick.sullivanflock.com/olympic.mp3>

Accept ultimate responsibility

- 10xers solve the hardest problems
- Inspire confidence that if you are “on it”, it will be taken care of, without follow-up
- You don’t get promoted and then get responsibility - you do the job and then you get promoted



Integrity and Trust

- Do what you say you are going to do
 - Set proper expectations
- You said it was done, is it really?
 - Have you used product as a potential user?
- You said it was fixed, is it really?
 - Have tests that give you high confidence
 - Verify yourself - don't throw over the wall

Expert Tooling

Increase development velocity

Investments in dev environments

The quality of a development environment is a reflection on the quality of your team - measured by how long a new hire takes for first commit/deploy. Re-reference: Simplicity

Broken development environments are drop-everything events, because they affect the productivity of the team

Work should be fun - like a video game

Pro tip: Be a multiplier by improving dev environments

Editing

- Invest in your editor - it is the tool of your trade. Sharpen and refine.
- Static Analysis in real time
 - Check on every file write
- Changes should be viewable immediately
 - Invest to reduce cycle time
- Pro tip - pre-commit linting to ensure consistency. Quality through tooling.

Debugging



Debugging

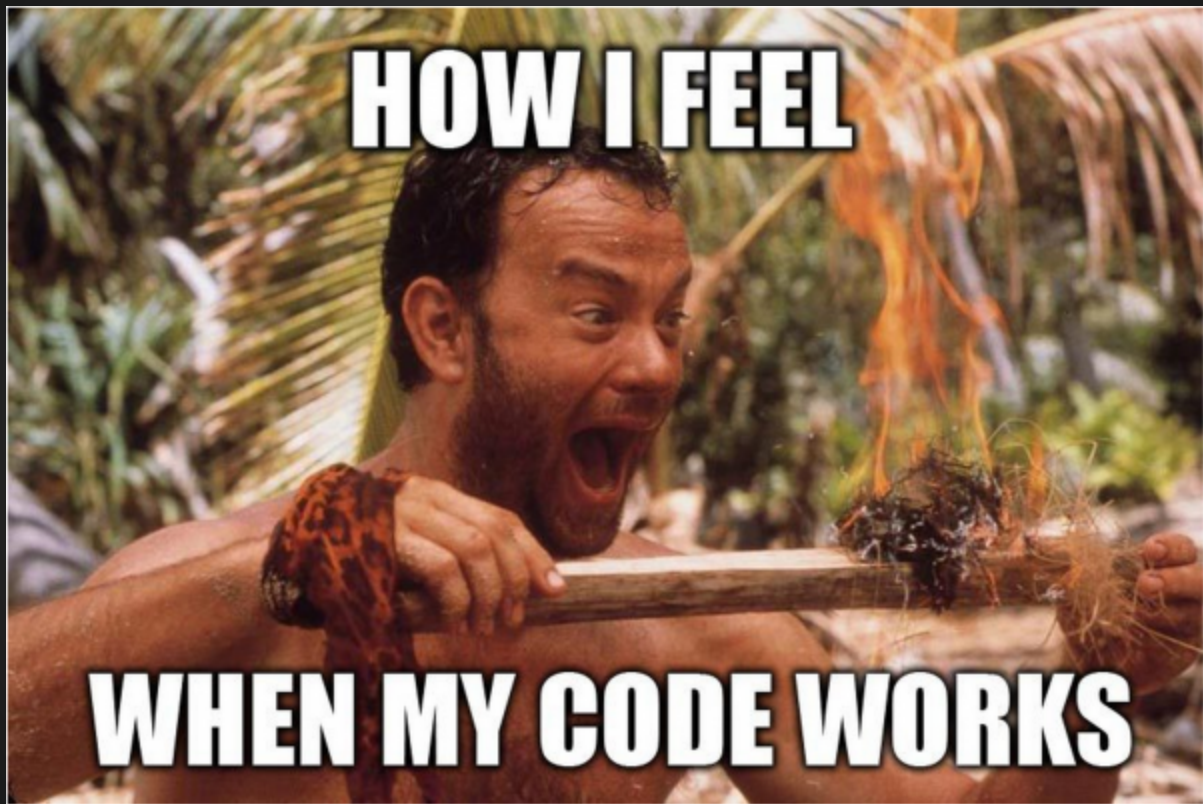
- Most of our time is spent debugging. Become an expert debugger (solve the hardest problems)
- Reproduce the problem
- Lessons from mechanics
 - Isolate and Test
 - Change one thing at a time
- Keep iteration time low - invest in tooling
 - Extra logging with a flag
 - Reporting of errors from production

Continuous Deployment

- Reduced iteration time == increased velocity
- Investments in development process will pay off forever. Step 0 of a new project is setting up Continuous Deployment workflow
- Continuous Deployment requires a higher level of rigor for testing - it's good for quality *and* productivity

Good tests

- Make others comfortable making changes to your code with good tests
- Good tests allow for *merciless refactoring*
- Don't punish bugs - punish lack of tests
- Separate release from deployment with feature flags. See: Django Waffle



Focus

Attention Quotient

The problems we are solving are hard. They require load time, and the smallest distraction can interrupt flow and force a reload.

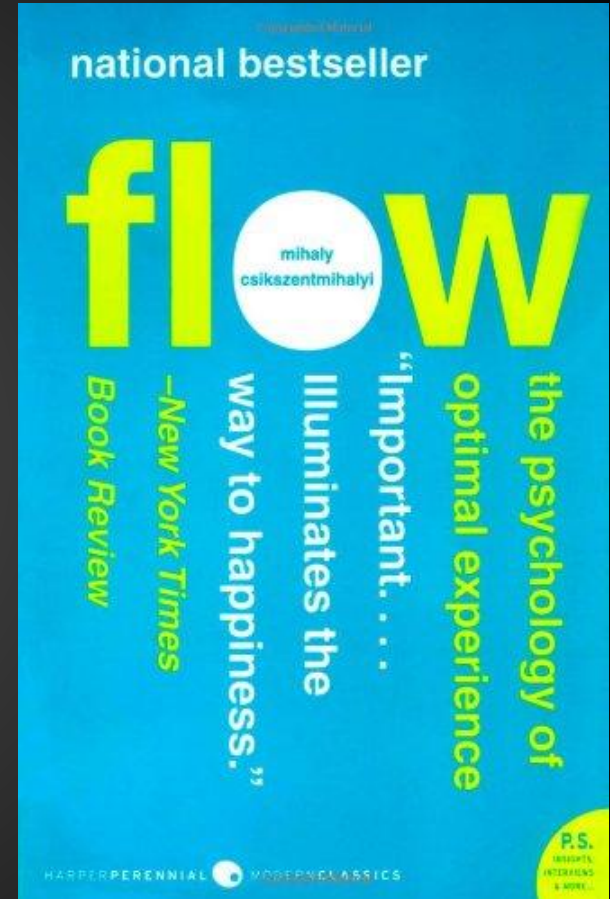
Your success depends on your ability to guard your focus. Aggressively defend it.

[Cartoon version](#)

Flow

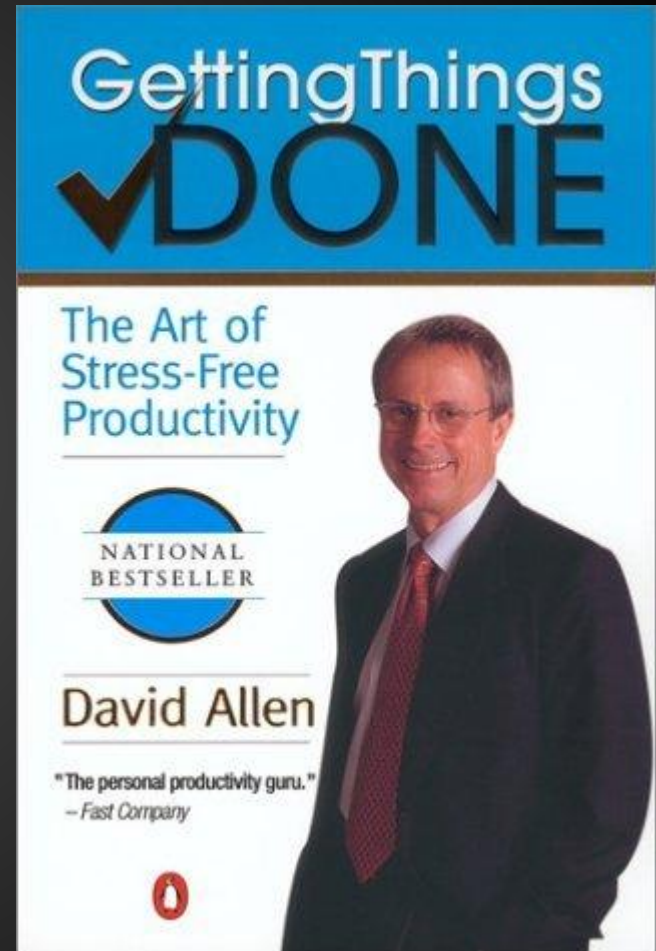
The efficient movement around obstacles

- Optimal state of productivity
- Get there
- Stay there



Getting Things Done

- Key point - separate *task management* from *doing*.
- Block off time for coding that is uninterrupted
- Good project managers provide this buffer, help you stay focused, and *multiply* your efforts



Computer Setup



- Turn off desktop notifications. Seriously.
 - Hide the dock so you don't see new messages
 - Get rid of interrupts
- RescueTime for Mac
 - Log time spent
 - Block distracting sites

Office Hacks

- Educate on the importance of focus
- Allow 1 high-interrupt channel (shoulder tap)
- Noise cancelling headphones
 - Office rule - headphones on == no interruptions
- Separate the talkers from the non-talkers
- Schedule meetings in groups - leave large blocks of uninterrupted time.
 - Pro tip - scrum right before lunch

Company email management

Hunt down and kill noisy emails

Champion good email habits

- <http://three.sentenc.es/>
- Informative, searchable subject lines
- [ACTION REQUIRED] prefixes
- Proper CC/Bcc

Our Brains

Understanding and Hacking

Different brains

- Left brain vs right brain
- Heart vs head
- Brain vs Gut
- Conscious vs subconscious

All are different explanations of the way our brains have evolved - often called the triune brain. Lizard, Mammal, and Homo Sapien.

Lizard Brain

The four F's.

- Food
- Flight
- Fight
- Sex

The mammal brain

- Relationships
- Empathy
- Passion/Drive

Homo Sapien brain (neocortex)

- Introspection
- Logic
- Delayed Gratification

Understand which brain to use

- Geeks think more with the homo sapien brain
 - Good for code, bad for interacting with people
 - Geeks aren't usually good managers
- User Experience, Creativity, and Meetings (listening) are better with mammal brain

Understand it and Hack it

Sleep/Time of day

- Pay attention to when you are the most creative versus focused
- Sometimes tired is better (lower creativity)
- Ray Kurzweil dreams ideas, then spends his days making them real
- Master the power nap. Seriously.
 - Hypnosis Downloads

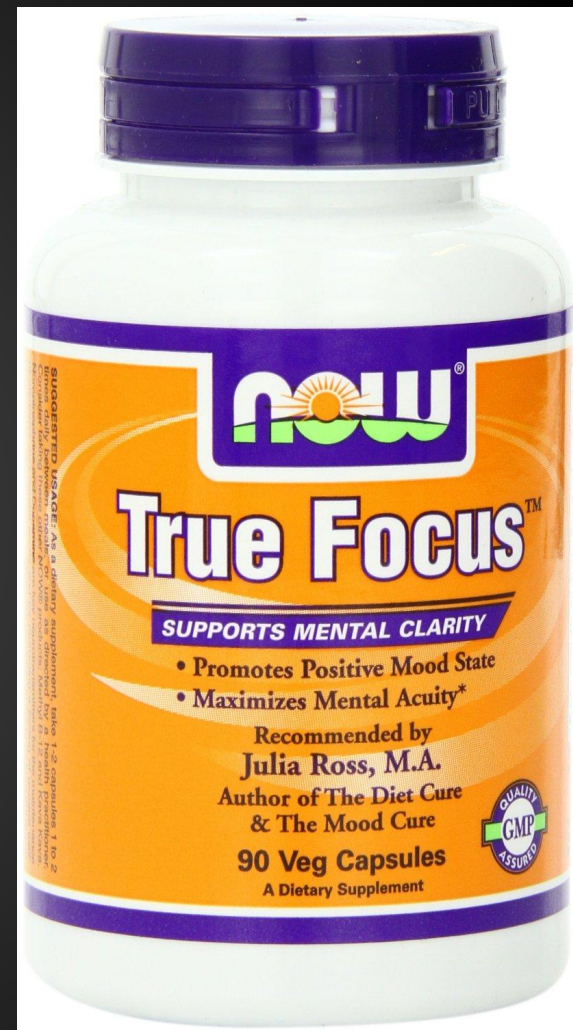
Showering

- Besides the obvious office morale reasons
- Causes different flows of blood to different parts of the brain.
- Combine with doing it first thing in the morning when you just wake up - best ideas of the day

Brain Health

Your brain is a temple

- Proper nutrition
- True Focus
- Essential fatty acids
 - Sushi!

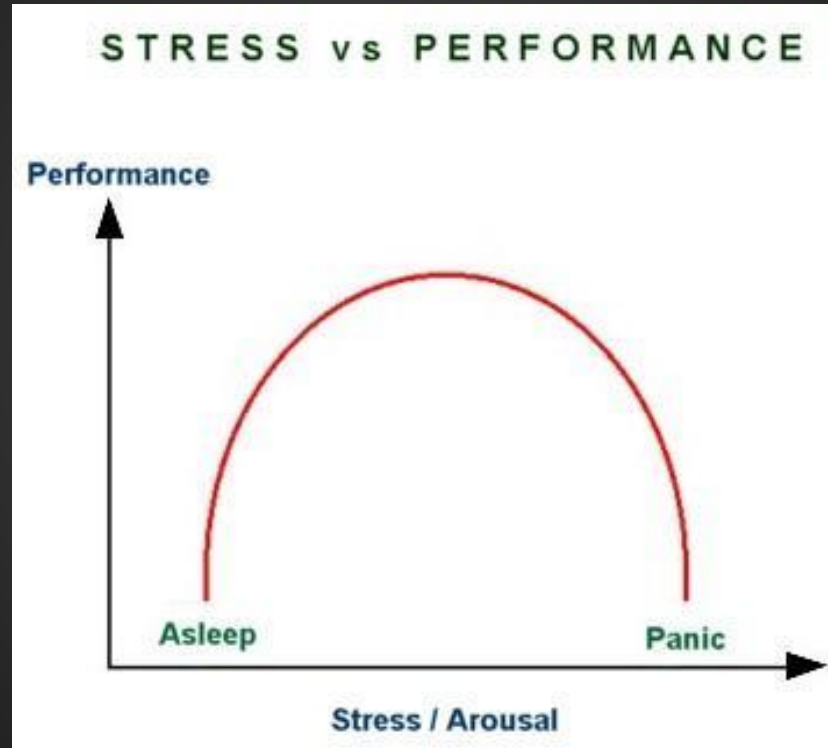


Brain Hacks

- Buzz - Science of Alcohol and Caffeine
- Caffeine
 - Good for logic. Bad for creativity. Not too much.
 - Situation dependent & men vs women
- Alcohol/being hungover (ballmer peak)
 - A little - good for focus & creativity
 - A lot - bad for productivity, great for stories
- Marijuana
 - Good for creativity; for some it improves focus

Me personally: alcohol + caffeine at night

Stress - good and bad



Be Happy

We are the most productive when we enjoy our work.

Change the circumstances - accept responsibility for making it work and go do it.

Go forth and Multiply

Questions? nick@sullivanflock.com

Twitter/Github/Everywhere: @gorillamania

Slides available at <http://creationeer.me/10x>

Links

- [The Mythical Man-month](#)
- [Flow](#)
- [True Focus pills](#)
- [Buzz - Science of Alcohol and Caffeine](#)
- [README-driven Development](#)
- [The Cathedral and the Bazaar](#)
- [Getting Things Done](#)
- [How to say it at work](#)
- [My homedir settings](#)