SOFTWARE ENGINEERING
IN STARTUPS

Dr. Vadim Zaytsev
Universiteit van Amsterdam
20 January 2014
CC-BY-SA
Introduction

- Vadim Zaytsev
- aka @grammarware
- currently at UvA
- worked at CWI (Amsterdam)
- worked at Uni Koblenz (Germany)
- studied at VU (Amsterdam), UTwente (Enschede), ...
- software language engineer
  software engineer
PART I
Startups

- Smaller companies
- Active software engineering
- Challenging & healthy environment
- Sufficiently different culture
Progress

- Horizontal
- Globalisation
- 1 to n
- Vertical
- Innovation
- 0 to 1

Why startups

- Size matters
- Coordination costs
- Politics
- Incentives
- Doing work vs. signalling about doing it

Why startups?

- Money?
- Fame?
- Changing the world?
- Costs of failure?

Dot com history

- Internet & WWW
- Mosaic & Netscape
- Web presence
- Prefix investing (e-, .com, nano-, 2.0)
- All ended around 2000

Dot com consequences

- Grand visions stability & incrementalism
- Know everything be lean & experiment
- Always advertise no
- Social antisocial
- Sales product
- Rapid monetisation reinvestment
- Talk about the future shut up

Peter Thiel, Blake Masters, Party Like It's 1999?, 2012.
PART II
First mover vs. last mover

Rocket City Space Pioneers, Ancient Chinese Rockets.
First mover vs. last mover

Geoff Parsons, Newton Lewt, 2008.
First mover vs. last mover

- Being on the frontier
- Creating your own market
- Escaping competition

Inside a startup

- Talented people
- Long-term orientation
- Generative spirit
- No diversity (productive disharmony)

Organisation
Organisation

HIGH
ALIGNMENT
STRUCTURE

LOW
ALIGNMENT
STRUCTURE

Organisation

TOTALITARIANISM  IDEAL

DOG-EAT-DOG  ANARCHY

Technical side

- Lots of freedom
- Possible to find exotic stuff
- Ability to use/try new technologies
- Lots of learning
- Quick feedback loops, fast release cycles
- No bureaucracy / little process

Kudos to Justin Halsall
More technical details

- In-house vs. external
- Often hacky [yet understood] ways
- Hatred for too much infrastructure
- Scrum, Kanban, TDD... never entirely
- Short-term focus: deploy often
- Unstable environment

Often encountered stuff

- Version control
- Some form of agile (+planning poker, etc)
- Code review
- Incrementality / refinement
- Ad hoc modelling
- Continuous integration

Ad hoc modelling

It’s all informal

Just a wiki page + Comments/emails

Startup community

- Open community
- Meetups & events
- Everybody is helpful
- Easy to ask for advice/feedback

Kudos to Justin Halsall
Networking

- “Is it efficient?”
- No explicit black-tie dinners
- Always networking
- Keeping up

Kudos to Justin Halsall
Pitching

- You are smart
- You have an idea
- ...
- PROFIT!

How to pitch

- Have a clear goal
- Know your audience
- Make it simple
- Do a pre-pitch

Financial side

- Lower pay up front
- Growing with the company
- People only get happier up to $70,000

Political side

- No bureaucracy
- Equal politics
- Input is welcome
- Lots of responsibility

Kudos to Justin Halsall
How to get hired?

- Join a startup instead of creating one
- Fit
- Broad set of skills
- Get to know people
- ... (3 things mentioned before) ...

PART III
Summary

- Aim at startups if you want to do innovative stuff (0 to 1)
- Build up and maintain a broad set of skills
- Network and be aware of surroundings
- Be open to learn & collaborate
- Enjoy
See also
• Sources of information/inspiration
  • given on the bottom of each slide
  • kudos to Justin Halsall and all other folks at H&F
  • kudos to Peter Thiel & Blake Masters for CS183.
• Slides?
  • http://grammarware.net/slides/2014/startups.pdf
• Fonts?
  • Avdira — George Douros, Unicode Fonts for Ancient Scripts, 2009.
  • Finger Paint — Ralph Oliver du Carrois, 2013.
  • Wild Honey — Denis Sherbak, 2013
• Questions? Ask or email or tweet.