Wir digitalisieren Ihre Welt!

Lessons learned as professionel software engineer

Frank Pientka since 1980

Wer ist Frank Pientka?

Dipl.-Informatiker (TH Karlsruhe)

Verheiratet, 2 Töchter

Principal Software Architect in Dortmund

Applications & Portals

Business Line Digital Enterprise

Fast 30 Jahre IT-Erfahrung

Projekte, Veröffentlichungen und Vorträge

Mehr Qualität in Software, Netzwerker, Innovator

Frank Pientka, Dipl.-Informatiker

frank.pientka@materna.de

+49 (231) 5599 8854

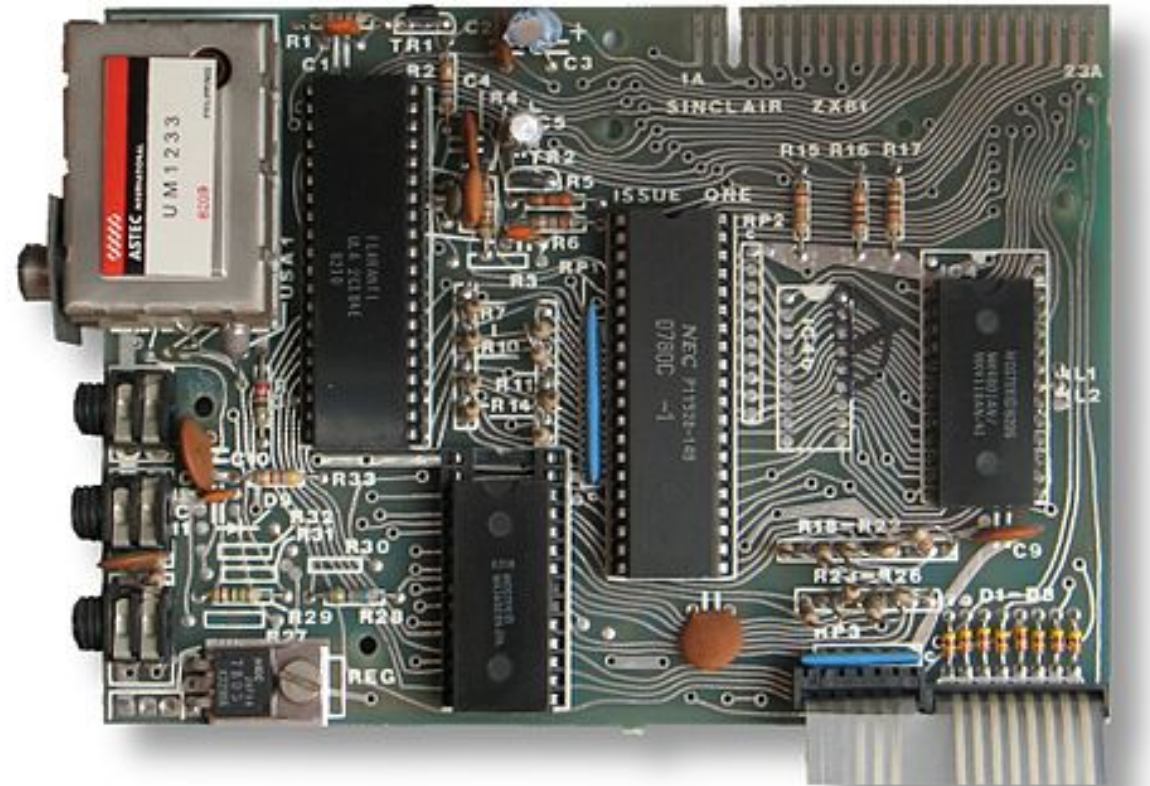 +49 (1570) 1128854

www.materna.de
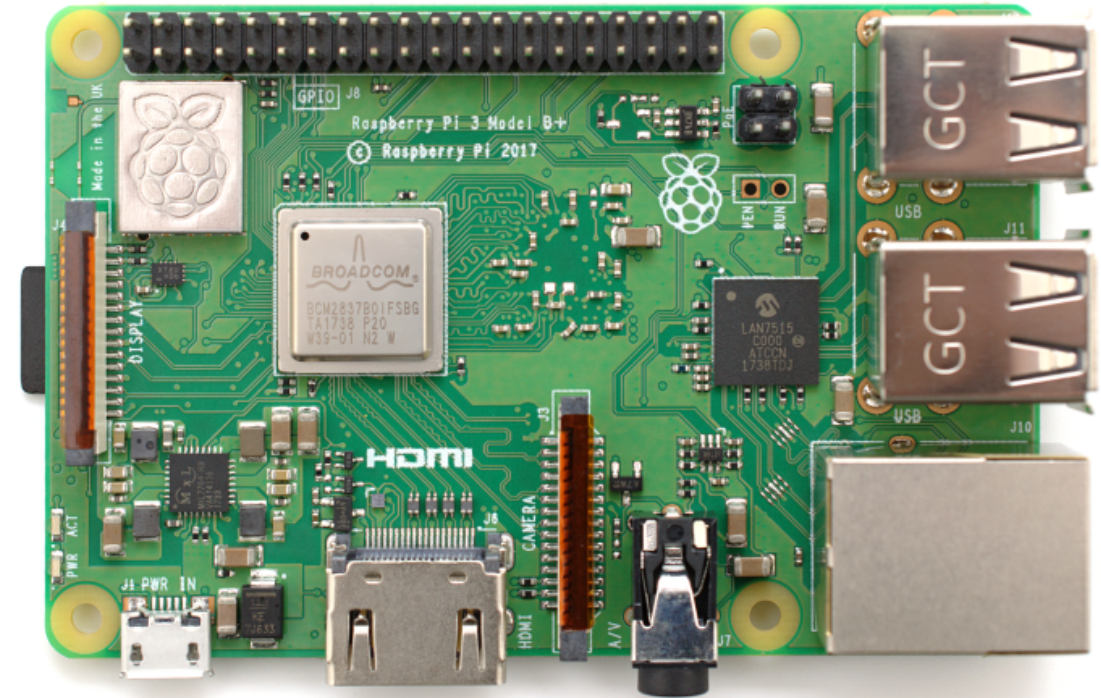
Fast 30 Jahre Berufserfahrung

1981 vs. 2018
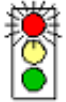
KB, MHz

GB, GHz, mulit-core

Das Internet früher: Texte statt Bilder, lang lebe ASCII

1992 Linux OpenSource

comp.os.minix ›

What would you like to see most in minix?

282 posts by 259 authors G+1

Linus Benedict Torvalds

Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)

   Linus (torv...@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(.

LINUX 0.12   BOOT DISK   /dev/hd4

LINUX 0.12   ROOT DISK

Root

/itsfoss   /itsfoss

Kathedrale oder Basar?
Wie wird Software entwickelt?

- hierarchical
- construction
- selected experts
- teams
- Perfection
- hidden
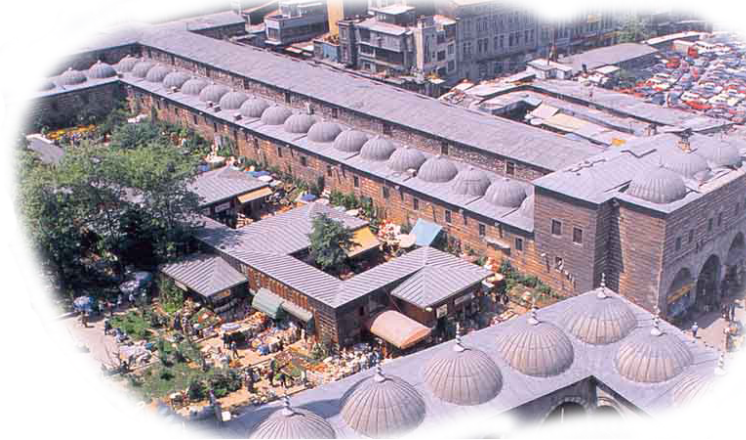
- network
- evolution
- peers
- community
- Improvement
- open

E.S. Raymond, the Cathedral and the Bazaar

THE CATHEDRAL & THE BAZAAR

"The most important book about technology today, with implications that go far beyond programming."
— Guy Kawasaki

MUSINGS ON LINUX AND OPEN SOURCE BY AN ACCIDENTAL REVOLUTIONARY

ERIC S. RAYMOND

WITH A FOREWORD BY BOB YOUNG, CHAIRMAN & CEO OF RED HAT, INC.
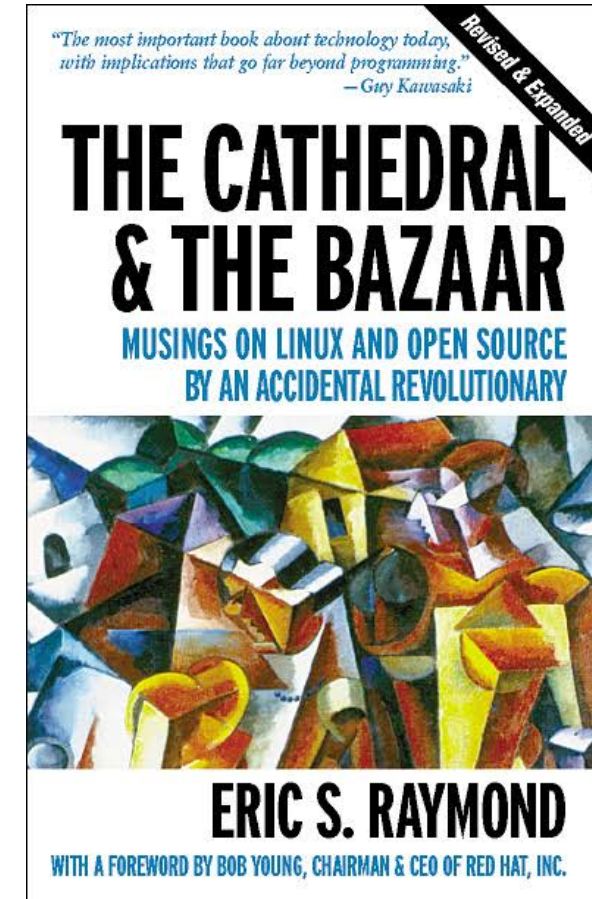
IN CONGRESS, April 1, 1787.

The agilist Manifesto of the thirteen united States of America.

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

peopleware!

17 man in a ski lodge february 2001

How Will Java Technology Change My Life?
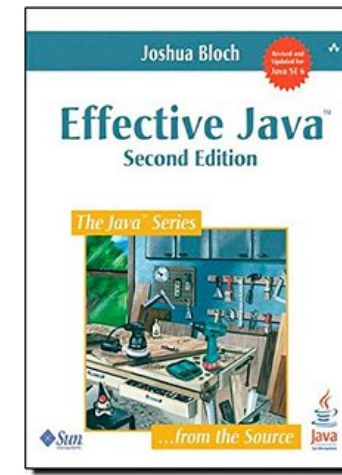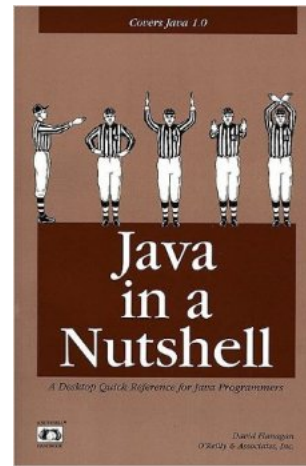
We can't promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

- **Get started quickly**: Although the Java programming language is a powerful object-oriented language, it's easy to learn, especially for programmers already familiar with C or C++.
- **Write less code**: Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program written in C++.
- **Write better code**: The Java programming language encourages good coding practices, and automatic garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans™ component architecture, and its wide-ranging, easily extendible API let you reuse existing, tested code and introduce fewer bugs.
- **Develop programs more quickly**: The Java programming language is simpler than C++, and as such, your development time could be up to twice as fast when writing in it. Your programs will also require fewer lines of code.
- **Avoid platform dependencies**: You can keep your program portable by avoiding the use of libraries written in other languages.
- **Write once, run anywhere**: Because applications written in the Java programming language are compiled into machine-independent bytecodes, they run consistently on any Java platform.
- **Distribute software more easily**: With Java Web Start software, users will be able to launch your applications with a single click of the mouse. An automatic version check at startup ensures that users are always up to date with the latest version of your software. If an update is available, the Java Web Start software will automatically update their installation.

JUnit Design (1998, Kent Beck, Erich Gamma): Sauberes Design

Design is still hard, but possible!

4 Klassen
6 Muster !
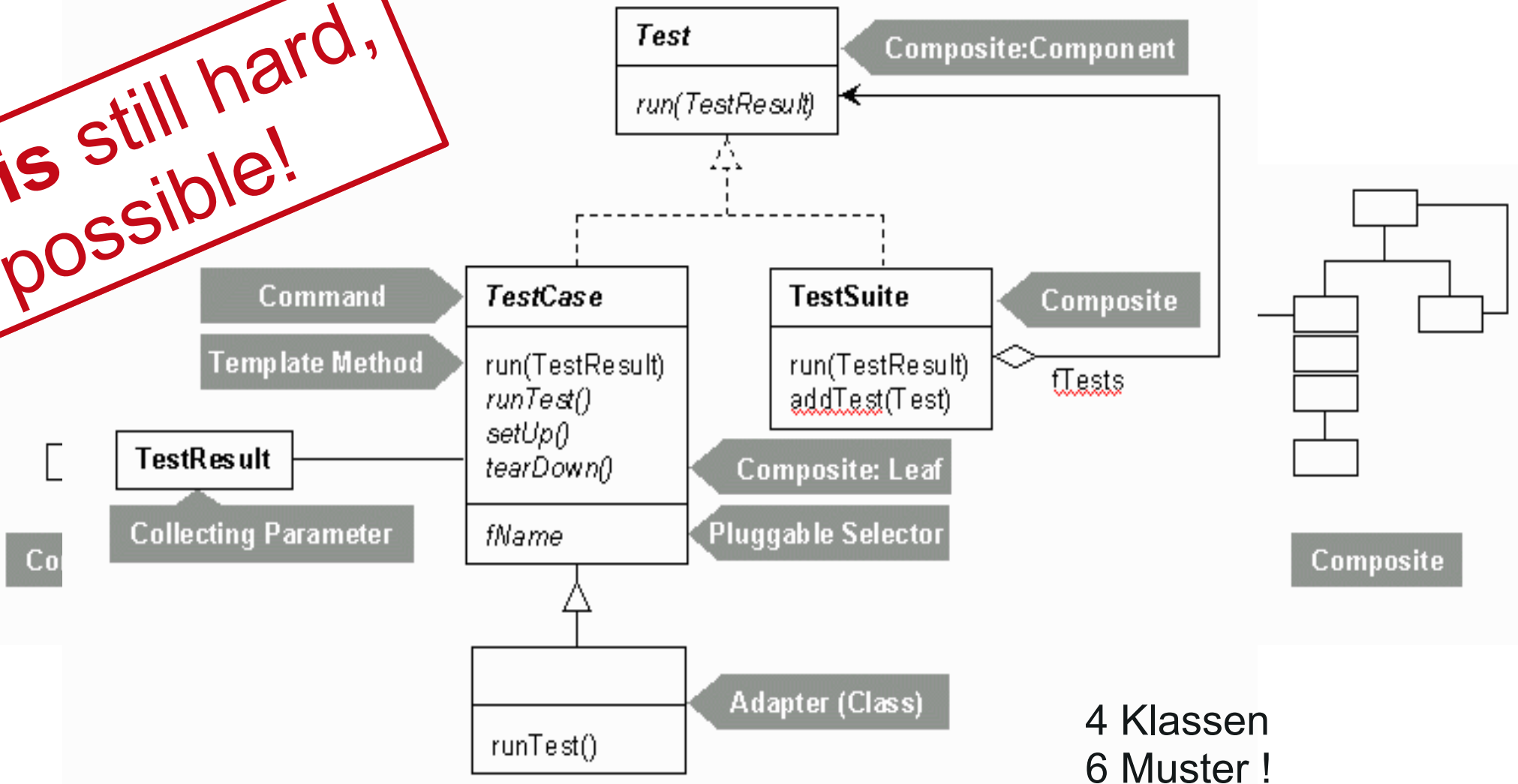
Test Infected: Programmers Love Writing Tests, Java Report, July 1998, Volume 3, Number 7
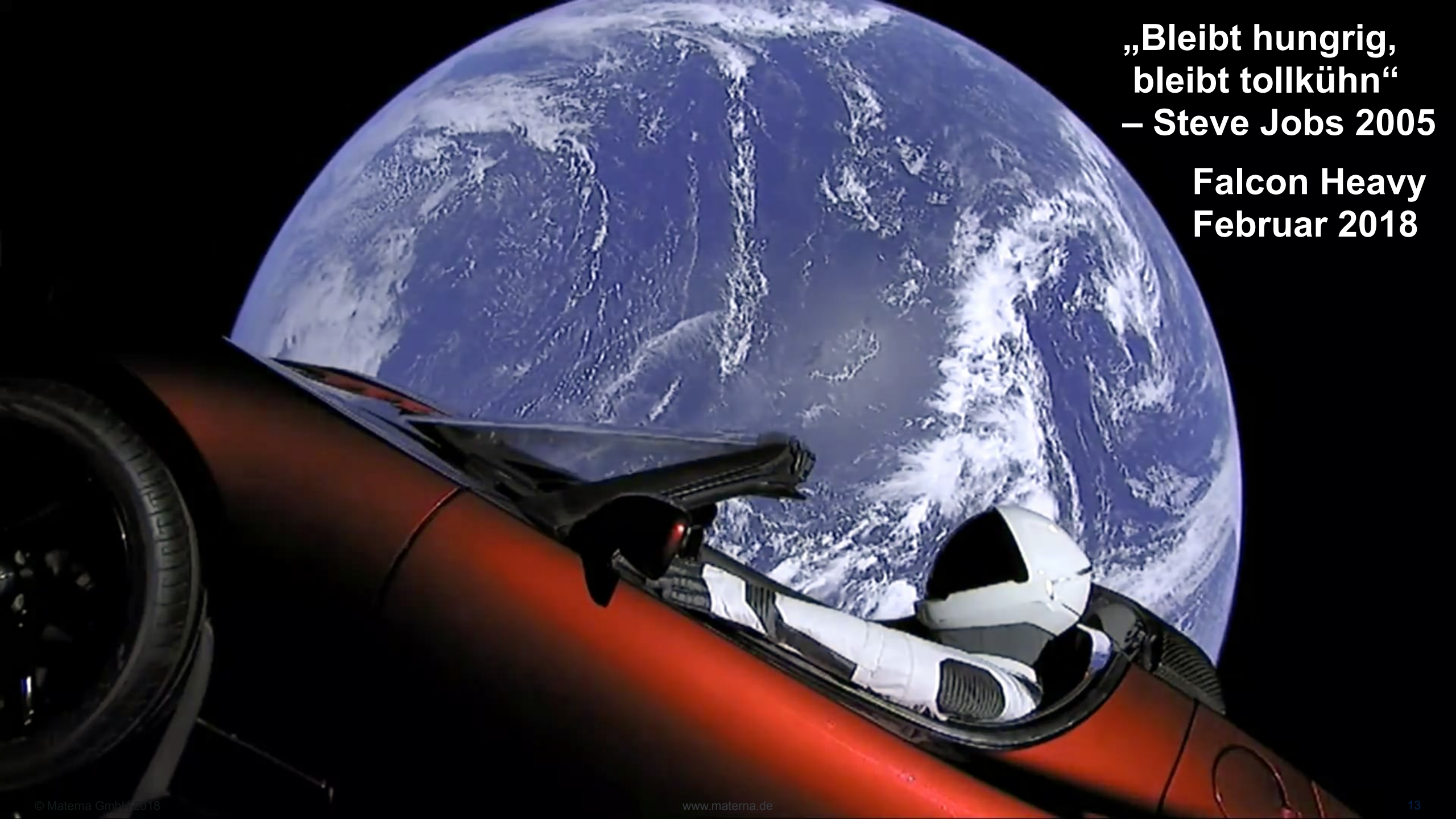
YOU BUILD IT

YOU RUN IT!

Fail Fast. Learn Fast.
Improve Fast.

– Spotify

„Bleibt hungrig, bleibt tollkühn"
– Steve Jobs 2005

Falcon Heavy
Februar 2018