Are we ready for the Software Factory?

An IT-Management Perspective

Hon.-Prof. Dr. Hermann Sikora
CEO, Raiffeisen Software GmbH
hermann.sikora@r-software.at

Executive Exchange 2023

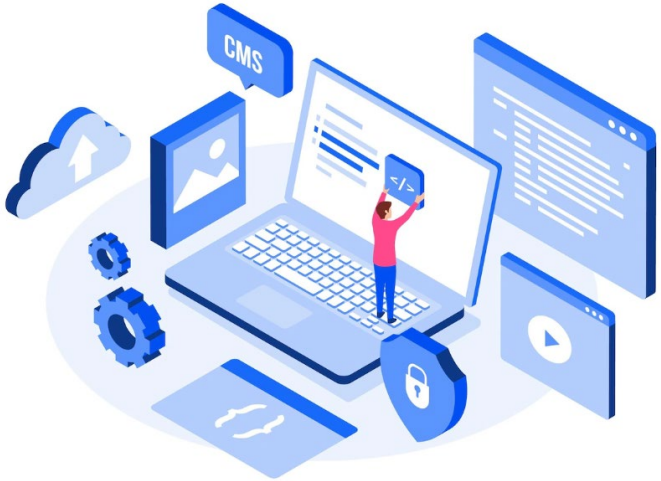
Navigating Software Production

Leading Software Factories to Excellence

March 29th, 2023

HPI Hasso Plattner Institut
Digital Engineering · Universität Potsdam


Software is eating the world.
Marc Andreessen
(2011)
quotefancy



There are many success factors for large companies.

Software is the second “**existence factor**” next to equity.

Software (engineering) is 100% business critical.

Image: Freepik

What do we mean by "factory"?

Features of handicraft enterprises

- Personal involvement and cooperation of the owners
- Sales order production instead of market production
- Manageable business volume (orders, employees, tools/machines)

Image: Freepik

The historical way to the factory

- **Handicraft business:** (tool-assisted) craft

- **Manufactory**: "More factory than craft features"
  - Specialization, division of labor, series production
  - (manageable) use of machinery

- **Industrial plant / factory**: "Real factory: high machine use, no handicraft"
  - Scalable, machine-based series production
  - Automation
  - Planning and control essential

Image by mudassir101 on Freepik

The historical way to the factory

- **Smooth transition from manufactories to factories (from 19th century)**

  => MATURITY / FITNESS topics:

  - Degree of machine use
  - High capital investment, great need for know-how

- **Handicraft enterprises in many branches up to date (example: carpentry)**

  => MATURITY / FITNESS topics:

  - Modernization and specialization related to craft performance
  - Regionality, personal relationship customer <-> craftsperson

Image by mudassir101 on Freepik

Features of the "historical factory"

19th. Ed., 1988

- Production of semi-finished and finished products in large quantities
- Use of specialized machinery
- High capital investment → High utilization of machine capacities
- Operating resources
- Larger number of specialized skilled workers in division of labor
- Production techniques determine workflow
- Systematic work preparation
- Production process with partially or fully automated operations

Features of the "Factory of the Future"

19th. Ed., 1988

- Production of semi-finished and finished products in large quantities
- Highest possible degree of automation
- Few "control persons", no (intensively) physically working people
- Process control systems and robots not only for production, but also for material transport, supply and removal of tools, and monitoring/control of product quality, machines and equipment
- Fully computerized manufacturing, digitally controlled machine tools
- Automation chain extends beyond actual production (pre- and post-phases: design, derivation of manufacturing specifications, customer quotations, sales, service).

=> REALITY TODAY

Historical development

HANDICRAFT

ALPHABETIZATION

MANUFACTORY

„VERLAGSSYSTEM“
(HOME WORK, „DECENTRALIZED PRODUCTION“)
„Home-Office since the 14th century.“ ☺

INDUSTRIALIZATION

FACTORY

AUTOMATIZATION

„FACTORY OF THE FUTURE“

DIGITALIZATION

ANALYTICS &
MACHINE LEARNING

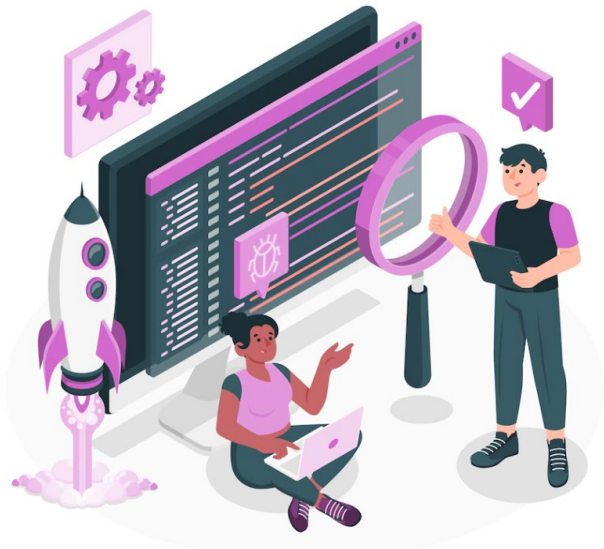
„SELF-ORGANIZING FACTORY“

Image by mudassir101 on Freepik

Image by zirconicusso on Freepik

S-o-t-A Software engineering has always tried to create the conditions for the "Software Factory"

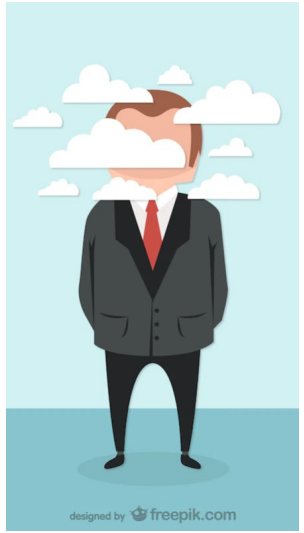
- **Reuse** of artifacts of any kind
  - Code, modules, templates, design patterns, development plans, ...
- Evolution / modernization of **development processes**
  - from "Chief Programmer Team" via "Waterfall" via Prototyping via ... to DevSecOps agility to ??
- (Process-) **Automation**
  - Testing, Building, Releasing, Rollout, Monitoring, Optimizing
  - Coding by generative AI (?)
- **Analytics & Metrics** (static and dynamic)
  - for software and software architectures as a basis for decisions



Images: Freepik: Oops icons by Flaticon; juicy_fish; storyset

Examples of inertial forces against the industrialization (factory orientation) of software development

- „Not invented here“ => (frequent) Mindset-Problem
- „Too big for us“ => can be true (for SMEs)
- High-specialist attitudes => sometimes justified
- Lack of know-how (dev.s) => can be remedied (if rcgnzd.)
- Investment aversion => biggest single problem
  („Analytics doesn`t write code“)
- Lack of know-how (dec. makers) => always dangerous



Difference between industrial manufacturing (cars, pharma, consumer goods, ...) and software development

- **Manufacturing of material (industrial) goods => three main processes**
  - Design and development of a prototype for serial production
  - Development of industrial serial production
  - Industrial series production ("multiplication")

- **Software development => one (iterative) main process**
  - Design, development, test, build, release, rollout, monitoring
  - No series production ("material multiplication") required

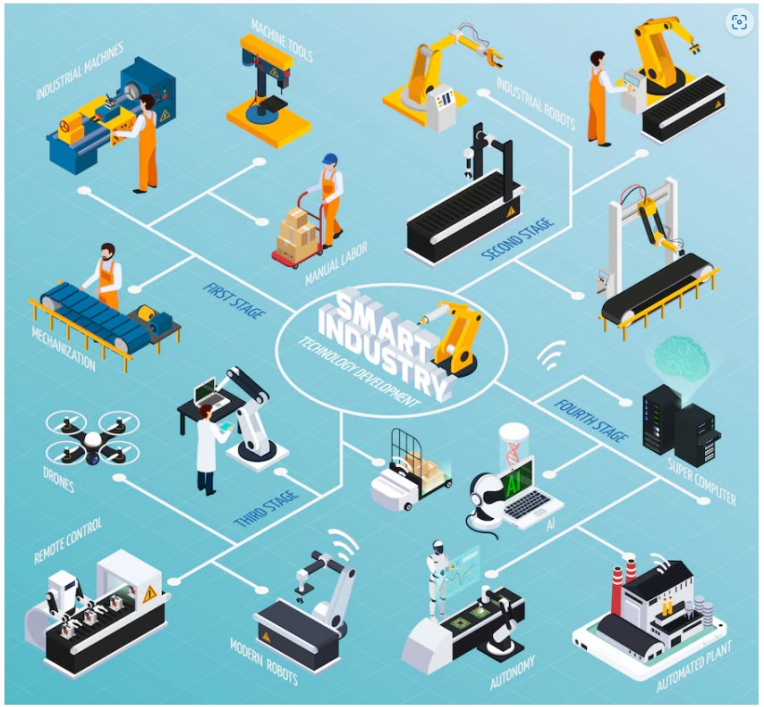


Image by macrovector on Freepik

Which characteristics of a "factory" should also be found in a "software factory"?

- Engineering definition ...
  - of all processes and artifacts
  - of proper use of all tools and semi-finished products
- Transparency, measurability, controllability
- Availability of static and dynamic parameters of all artifacts
- Assured repeatability
- Continuous improvement process

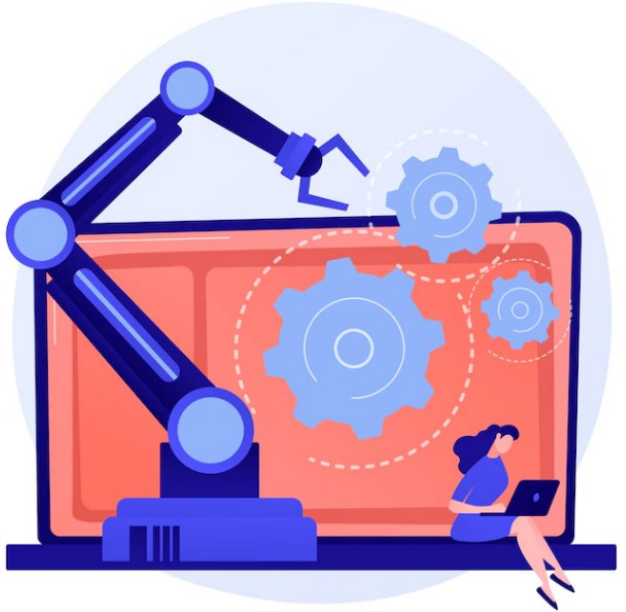


Image by vectorjuice on Freepik

Software development has long since "diversified" - from craft to industrial scale. But does this also mean that "software development on an industrial scale" today is actually "factory-industrial", i.e. “seamless engineering”?

Image by jcomp on Freepik

„Development-in-the-small“

- Manageable scale, relatively simple requirements
- Often "scripting" or based on low/less code platforms
- Mostly for customizing standard software
- Or in the environment of standard software and their API worlds
- In companies (SMEs) that typically do not have or need a (full) software development lifecycle

Image: Freepik

„Development-in-the-large“

- Full Scope Software Development Lifecycle – „all-in“
- Large companies in all industries that require in-house development as well as standard software
- Manufacturers of standard software of all types
  - Industry software suppliers of all sizes
  - Big global players are massive drivers of the industrialization of SWE - all others are massively challenged not to lose ground

Image by mudassir101 on Freepik

What are the drivers of the ongoing topic of "industrialization of software development"?

- Complexity of increasingly powerful technology stacks
- Communities and (open source) resources
- Fear of no longer being able to adequately manage the overall complexity
- Easy access to cloud services
- Scaling needs
- Increasing amount of legacy software
- Advances in analytics tools (static, dynamic)
- Analytical & generative AI in software development
- Security engineering

Image by fullvector on Freepik

Results-oriented management

is not possible without **transparency,**

which would remain **actionism.**

What is transparency in the present context?

Images: katemangostar, macrovector on Freepik

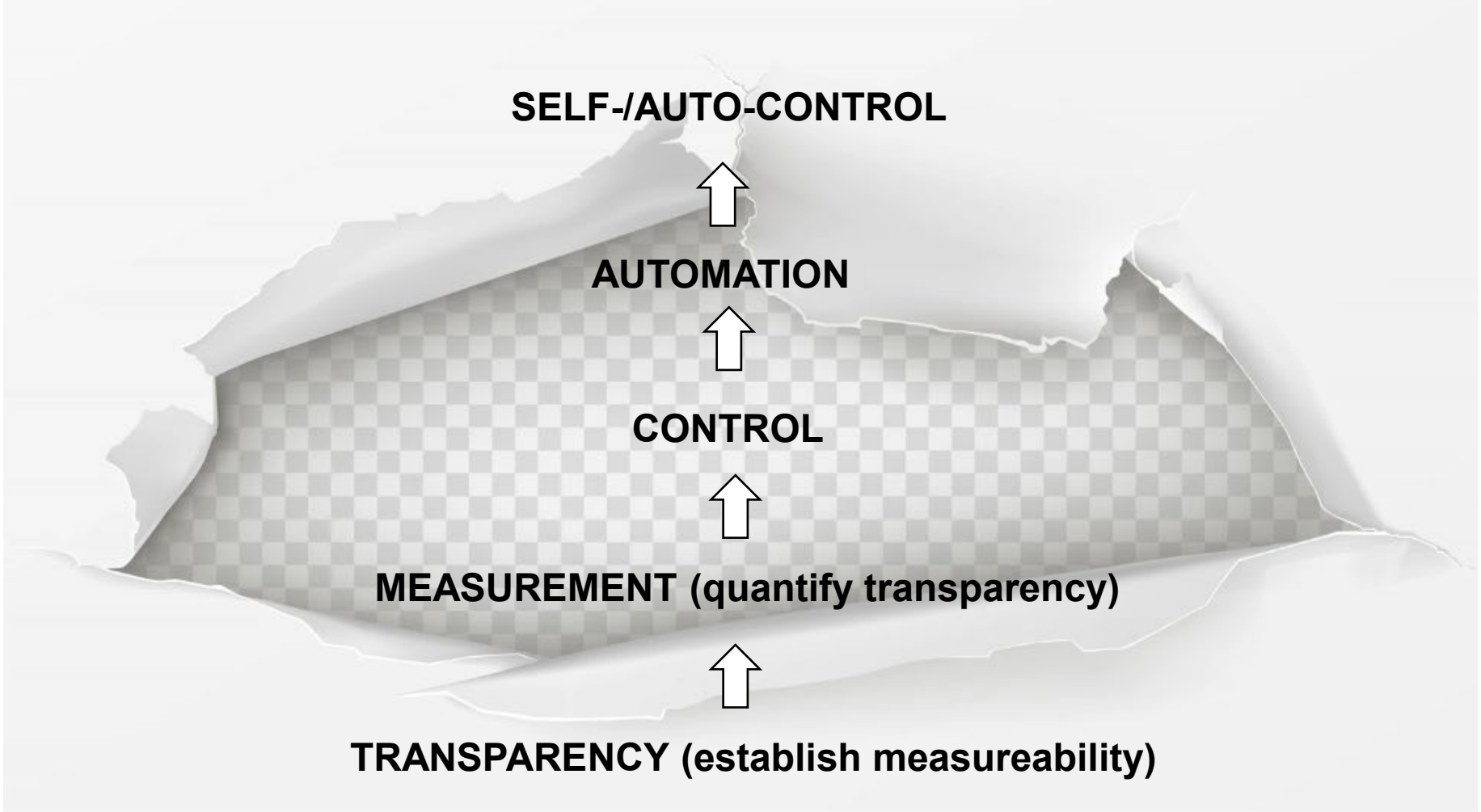


Image by vectorpocket on Freepik

„Software Factory“

Product-suite of a manufacturer

„Adapting, assembling, configuring“ based on a „factory scheme“

≠

Organizational form of a process defined in engineering terms

Development and maintenance of software (transparent, measurable, controllable, assured repeatable, constantly improving)

"Software Factory" in the meaning of "Product Suite of a Manufacturer"

Product-oriented "Factory Scheme" for "Adapting, Assembling, Configuring" with:

- Framework(s), Product-Template(s), Process-Template(s)
- Domain-specific language(s)
- assistance systems for code generation
- various tools, repositories, APIs, ...

=> vendor-specific, possibly domain-specific

=> optimized for certain classes of applications

=> broad target market, also suitable for SMEs

=> "real" software development only for new (missing) components

Image by vectorjuice on Freepik

"Software Factory" in the meaning of "engineering process"

Organizational form of an engineering-defined (=measurable, controllable, assuredly repeatable, constantly improving) process for the development and maintenance of software

- Design, coding, testing, building, releasing, monitoring processes
- Knowledge, tools, templates, code of many communities / manufacturers / domains
- Cloud technologies for infrastructure abstraction
- Agility, DevOps, DevSecOps, ... with (partial) automation

=> vendor-neutral, domain-neutral, universally applicable

=> Target market: companies that develop business-critical software and need vendor- and product-neutral engineering

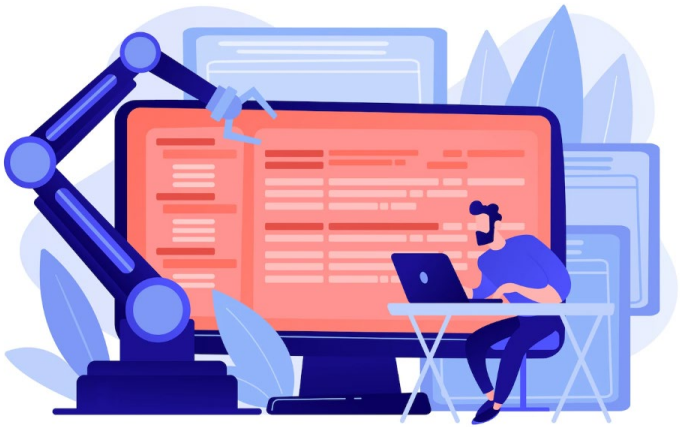


Image by vectorjuice on Freepik

Software has eaten the world

Posted on December 11, 2017 by Ed Pearson - *3 Minute Read*

Why don't have all companies a mature Software Factory yet? (1/2)

- **A Software Factory is (currently) the Champions League of Software Engineering**
  - requires high capital investment (factory!) and corresponding know-how; laborious establishment
- **Glaring shortage of (very) well educated (business) computer scientists**
  - at all levels; no improvement in sight, on the contrary
- **Overriding importance of software (engineering) not completely clear to all C-levels**
  - relatively few "technology people" on boards (general know-how question) - in the digital age

Why don't have all companies a mature Software Factory yet? (2/2)

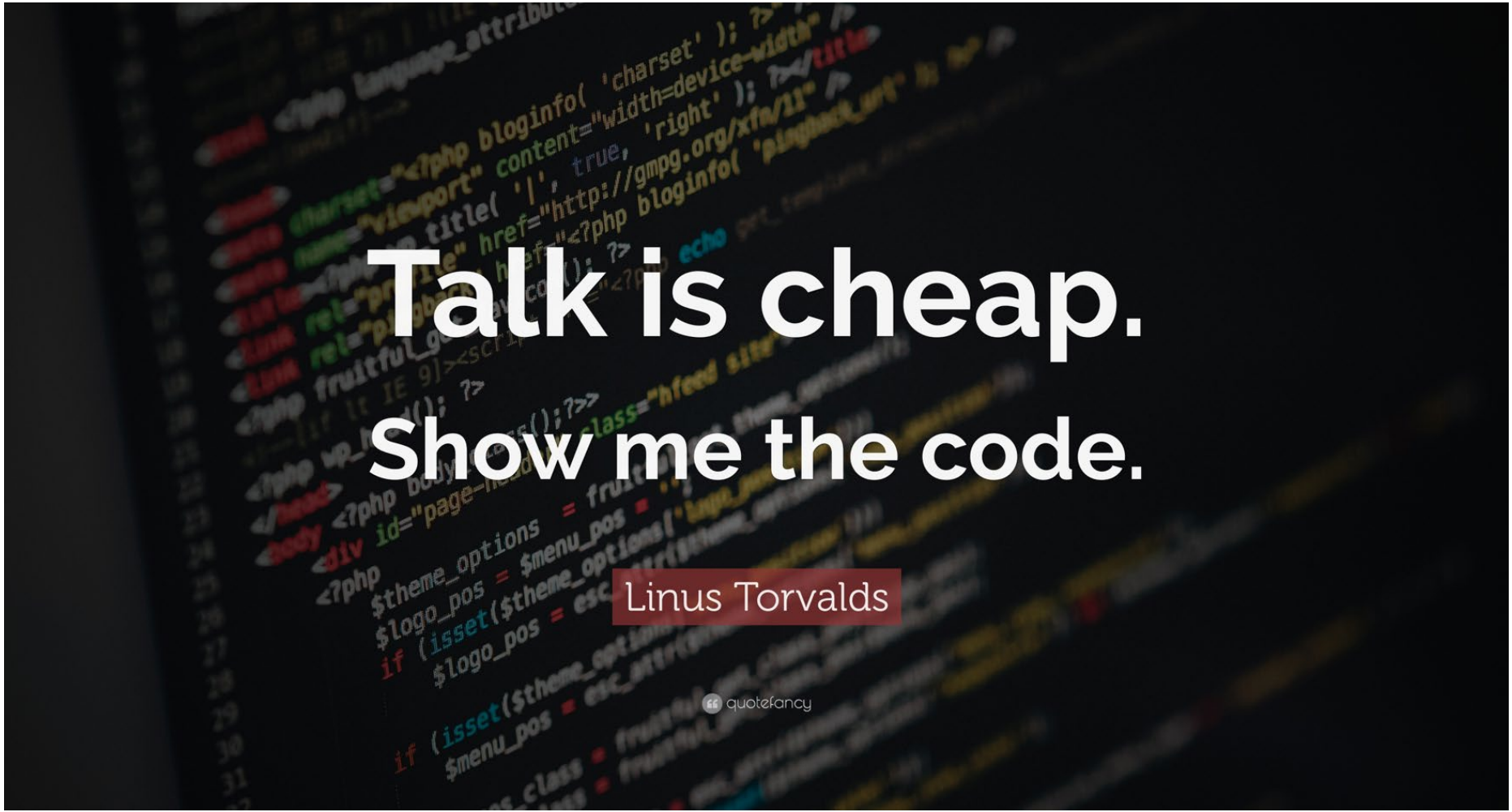
- **Consequence:**

  hardly any proactive management action "pro software engineering“

  - It is already seen as progress when software (engineering) is no longer seen as a cost factor (it is a survival factor) and "agile work is allowed“

- **Abstract factory topics such as "analytics" (and their benefits) are enormously difficult to "sell“**

  - "Analytics doesn't write code and is expensive - there is no budget for it."


Talk is cheap.
Show me the code.
Linus Torvalds
quotefancy

What do we do at Raiffeisen Software ?

- Agile development model
- Agile structural model (product-oriented organizational structure)
- Transparency:
  - Dashboard "Customer Portal" => mature and proven
    - Up-to-the-minute details on all phases and artifacts of all projects (drill-down)
    - Main target group: customers (all levels thereof)
  - Dashboard "Internal Single Point of Truth" => basis developed
    - various KPIs as input for management
    - main target group: development management; general management
  - various reporting engines (e.g. release automation)

”

What‘s next?

KI-generiertes Bild, der Prompt lautete (auf Englisch): »Büste, die in ihrem Kopf eine neue Cyberwelt bildet, der Hinterkopf und die Schultern bestehen vollständig aus Technologie, distanzierter Gesichtsausdruck, Kopf zur Kamera geneigt, detailliert minimalistisch, Farben orange, weiß und schwarz« DER SPIEGEL / Gestaltet mit Midjourney

FORBES > INNOVATION > AI

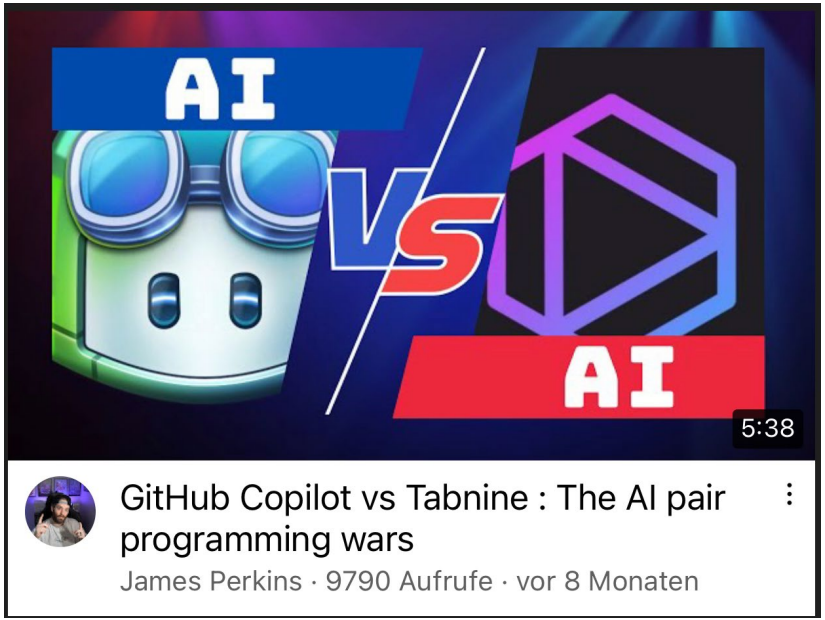
Software Ate The World, Now AI Is Eating Software

By Martijn van Attekum, Jie Mei and Tarry Singh

forbes.com,
Aug. 29th, 2019

Artificial Intelligence and the Software Factory

- Analytical and generative **AI will have a tremendous impact on software engineering**, in all domains
  - Analyzing, Testing, Coding, Building, Releasing, Monitoring, …
- In the future, **AI will form the basis of all engines in a software factory**

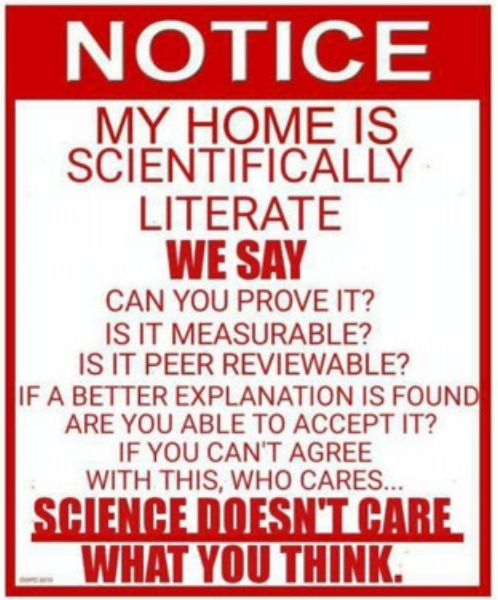


Conclusion:
“ Yes, we are able to implement a Software Factory, if we are ‘allowed to want it’ "

- **Software engineering always strives for factory approaches**
  - but establishing them is an enormously laborious and cost-intensive, but worthwhile undertaking
- **Software engineering has universal problem-solving competence - that is the "problem”**
  - it will continue to "diversify" - also by means of factory approaches
- **Every (large) company will have to choose its factory approach**
  - or one day find itself in a pure consumer role (with corresponding dependencies)
- **"Result-oriented controllability" is THE core benefit of a software factory**
  - Complexity remains controllable, "quantifiable transparency" is the management lever

Image by juicy_fish on Freepik

Image by Patrickss on Freepik



Note: This teaching/learning/lecture document contains topic-related product placements and company mentions, especially in references and examples. Errors and changes reserved. Any liability excluded.