Invention, Innovation and Diffusion in Software Development Techniques

Keith Braithwaite
© Zühlke 2010

Context

zühlke
empowering ideas

"I come as an entertainer, not as a salesman. I want you to enjoy these ideas because I enjoy them"—

Alan Watts

Keith Braithwaite
© Zühlke 2010

The Pessimistic Meta-inference

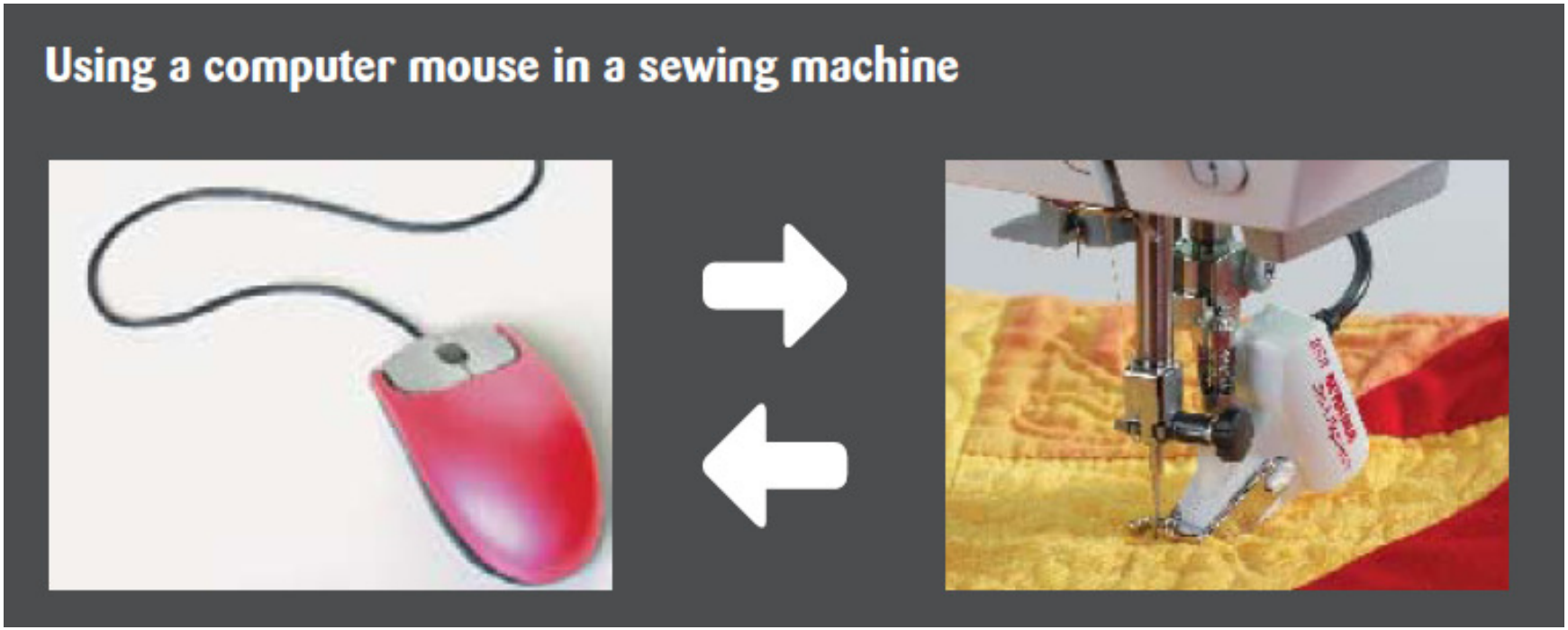
zühlke
empowering ideas

"if past scientific theories *which were successful* were found to be false, we have no reason to believe the […] that our currently successful theories are approximately true" –after Laudan, 1981 emphasis added

Keith Braithwaite
© Zühlke 2010

zühlke
empowering ideas

Innovation is different from invention



Existing ideas and technologies adapted to new purposes

Keith Braithwaite
© Zühlke 2010

Anti-innovation?

zühlke
empowering ideas

Is the idea of an XML build file innovative?

```
<!-- Compile target -->
<target name="compile" depends="prepare" description="Compile Java sources">
        <!-- Compile Java classes as necessary -->
        <mkdir dir="${classes.home}" />

        <javac srcdir="${src.home}" destdir="${classes.home}" debug="${compile.debug}" deprecation="${compile.deprecation}" optimize
                <classpath refid="compile.classpath" />
        </javac>

        <!-- copy over some .wav file resources -->
        <copy todir="${classes.home}">
                <fileset dir="${src.home}" includes="**/*.wav" />
        </copy>

</target>
```

Existing ideas and technologies *misapplied*

© Zühlke 2010

Diffusion

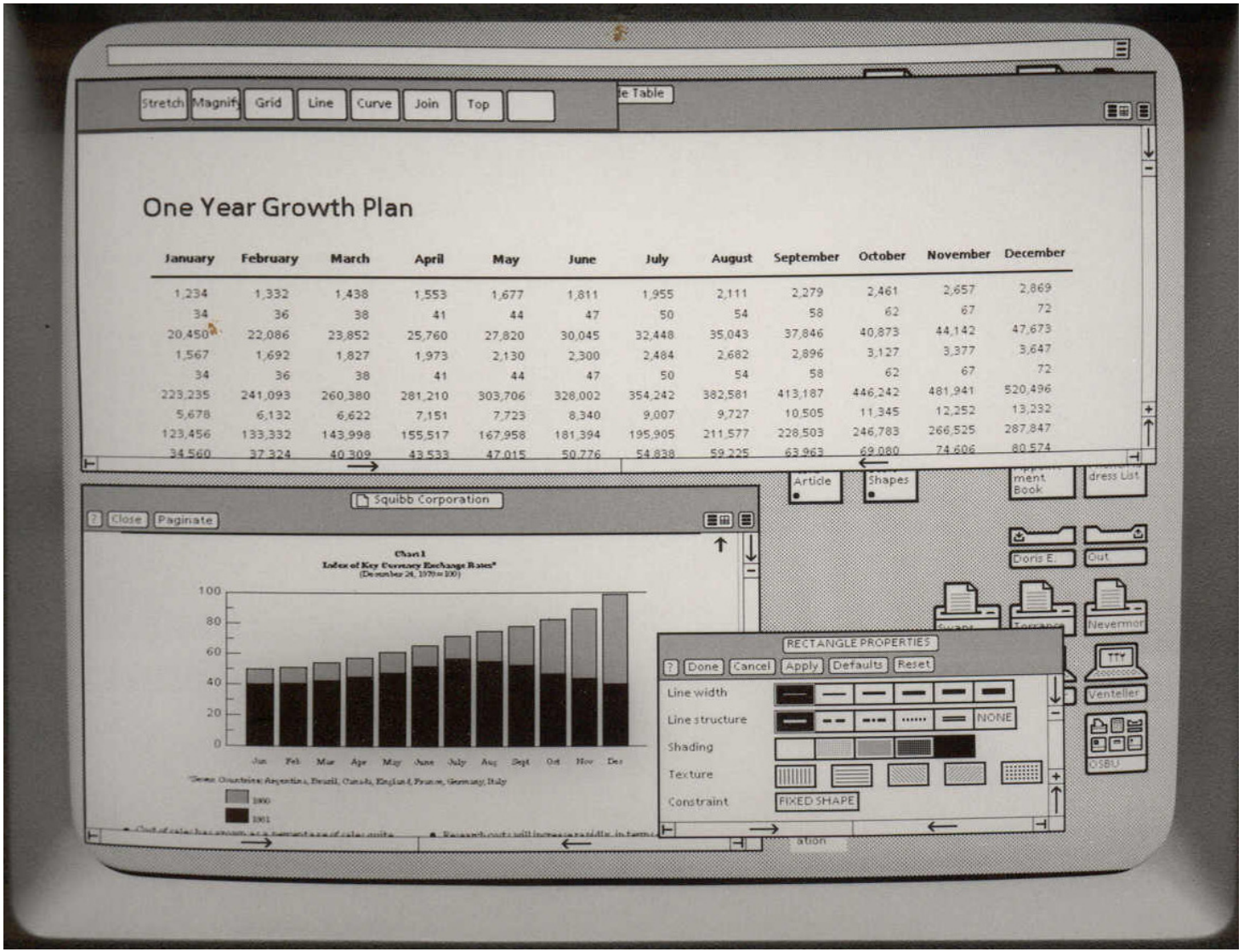
zühlke
empowering ideas

"The future is already here, it's just not very evenly distributed" –William Gibson, 1993

Keith Braithwaite
© Zühlke 2010

Xerox Star: 1981

zühlke
empowering ideas



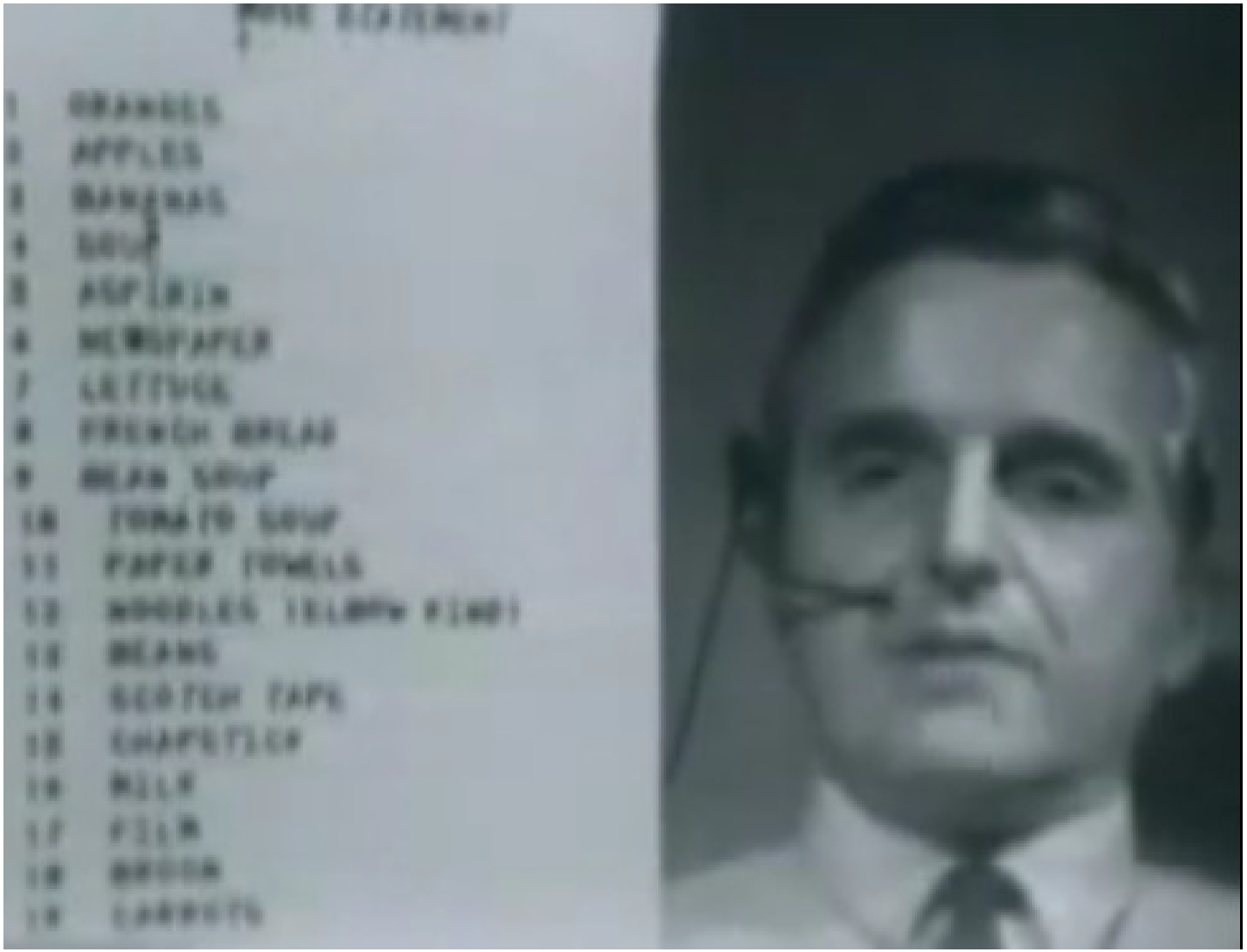
http://www.digibarn.com/collections/screenshots/xerox-star-8010/xerox-star-8010-02.jpg

Keith Braithwaite

© Zühlke 2010

SRI oN Line System 1968

zühlke
empowering ideas



Invention and Innovation
Invention, Innovation and Diffusion in Software Development Techniques
Slide 8
6 October 2010

Keith Braithwaite
© Zühlke 2010

Sources of Diffusion

zühlke
empowering ideas

PhD

- Rsync from Andrew Tridgell's thesis
- RESTful services from Roy Fielding's
- Etc…

Industry

- Sharding
-

Keith Braithwaite
© Zühlke 2010

Diffusion from Other Industries

zühlke
empowering ideas

EXHIBIT 1
Sequential (A) vs. overlapping (B and C) phases of development

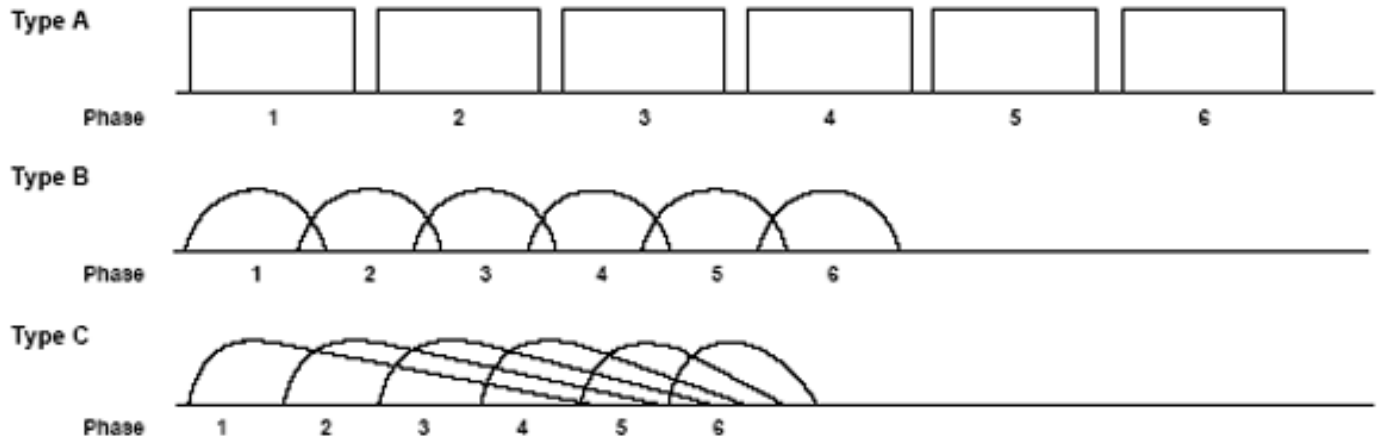


Exhibit 1 illustrates the difference between the traditional, linear approach to product development and the rugby approach. The sequential approach, labeled type A, is typified by the NASA-type PPP system. The overlap approach is represented by type B, where the overlapping occurs only at the border of adjacent phases, and type C, where the overlap extends across several phases. We observed a type B overlap at Fuji-Xerox and a type C overlap at Honda and Canon.

"The New New Product Development Game"– Takeuchi, Nonaka 1986

© Zühlke 2010

Scrum

Keith Braithwaite
© Zühlke 2010

zühlke
empowering ideas

Fully Distributed Scrum: The Secret Sauce for Hyperproductive Offshored Development Teams

Jeff Sutherland, Ph.D.
Scrum, Inc.
Boston, MA, US
jeff.sutherland@computer.org

Guido Schoonheim
Xebia b.v.
Hilversum, Netherlands
gschoonheim@xebia.com

Eelco Rustenburg
Xebia b.v.
Hilversum, Netherlands
erustenburg@xebia.com

Maurits Rijk
Xebia b.v.
Hilversum, Netherlands
mrijk@xebia.com

Abstract

Scrum was designed to achieve a hyperproductive state where productivity increases 5-10 times over industry averages and many collocated teams have achieved this effect. The question for this paper is whether distributed, offshore teams can consistently achieve the hyperproductive state. In particular, can a

of over 1M lines of code. A distributed team of over 50 people in the U.S. and Russia delivered velocity per developer equivalent to a collocated Scrum team and produced the same number of features as a typical 350 person waterfall team [7].

During 2006-2008, Xebia implemented a distributed software development team model on multiple projects of variable types with teams half in

http://jeffsutherland.com/SutherlandFullyDistributedScrumXebiaAgile2008.pdf

© Zühlke 2010

zühlke
empowering ideas

Fully Distributed Scrum: The Secret Sauce for Hyperproductive Offshored Development Teams

Jeff Sutherland, Ph.D.
Scrum, Inc.
Boston, MA, US
jeff.sutherland@computer.org

Guido Schoonheim
Xebia b.v.
Hilversum, Netherlands
gschoonheim@xebia.com

Eelco Rustenburg
Xebia b.v.
Hilversum, Netherlands
erustenburg@xebia.com

Maurits Rijk
Xebia b.v.
Hilversum, Netherlands
mrijk@xebia.com

Abstract

Scrum was designed to achieve a hyperproductive state where productivity increases 5-10 times over industry averages and many collocated teams have achieved this effect. The question for this paper is whether distributed, offshore teams can consistently achieve the hyperproductive state. In particular, can a

of over 1M lines of code. A distributed team of over 50 people in the U.S. and Russia delivered velocity per developer equivalent to a collocated Scrum team and produced the same number of features as a typical 350 person waterfall team [7].

During 2006-2008, Xebia implemented a distributed software development team model on multiple projects of variable types with teams half in

A peer-reviewed paper describing real work, just a few of issues…

© Zühlke 2010

Not a Secret

zühlke
empowering ideas

"Requests start to come in faster, and with more urgency. By the end of a few months, it takes half a day for me to even respond to all of them. Every request is an emergency. I get nothing done, and without much notice, programming isn't what I get to do anymore. I love writing software, but the work is unbearable. I could never stop seeing myself as a software engineer, but I'm wondering if the industry as I had envisioned it does not really exist."

—redditor deltnurgsid

© Zühlke 2010

Not a Secret

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this. […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

Keith Braithwaite
© Zühlke 2010

Not a Secret

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this. […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

Keith Braithwaite
© Zühlke 2010

Not a Secret

zühlke
empowering ideas

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this. […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

© Zühlke 2010

Not a Secret

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this. […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

© Zühlke 2010

Not a Secret

zühlke
empowering ideas

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but **I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this.** […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

© Zühlke 2010

Not a Secret

zühlke
empowering ideas

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this. […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

© Zühlke 2010

Not a Secret

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this. […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

Keith Braithwaite
© Zühlke 2010

Not a Secret

"Some advice from the world of book publishing […] multiple sales people all wanting different things, all bugging me over and over. I gathered them at a meeting and said that I'd be happy to help them but they had to decide amongst themselves which projects and features got priority […] when a salesperson came to me mid-week and pushed for something outside the list, I could say no and refer them to the list. They'd beg but eventually everyone got onboard with the manta of "If it's not on the list, don't even ask" […] I can work on that but I can only do it tonight after 5:30pm so I'll have to work overtime. If I do, then you have to be here with me until it's done." […] I never got a yes to this. […] Arrive on time and leave on time and take your lunch break every single day no matter what […] Make some unilateral decisions. This is the hardest one but it can also work incredibly well. I simply said no to some people […] Make it very clear to everyone in a single meeting that each request cuts away time and they will personally be responsible."– *redditor mathewferguson*

Keith Braithwaite
© Zühlke 2010

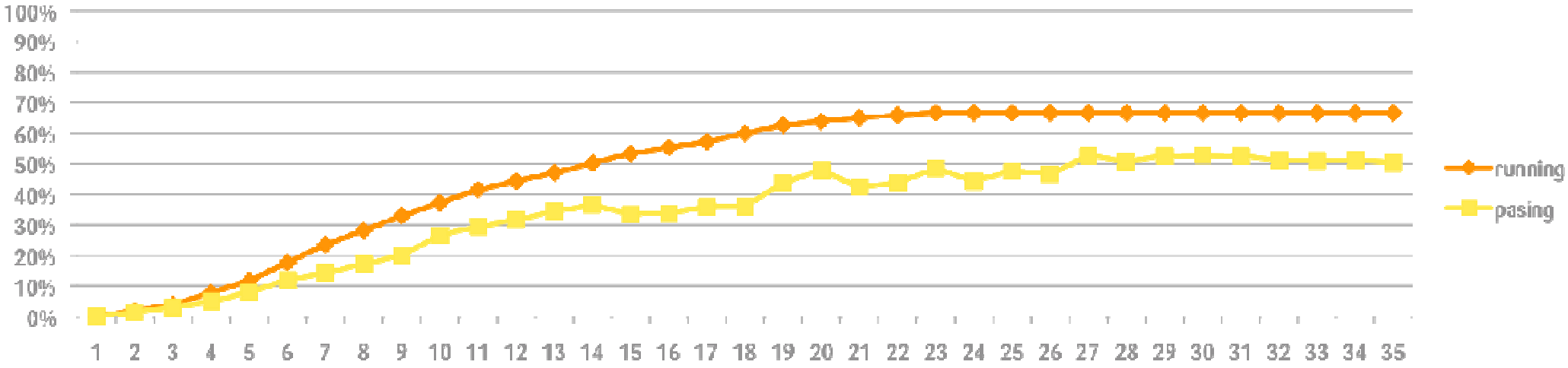
Hyperproductivity?

Keith Braithwaite
© Zühlke 2010

Hyperproductivity?

zühlke
empowering ideas

- Instead of this:



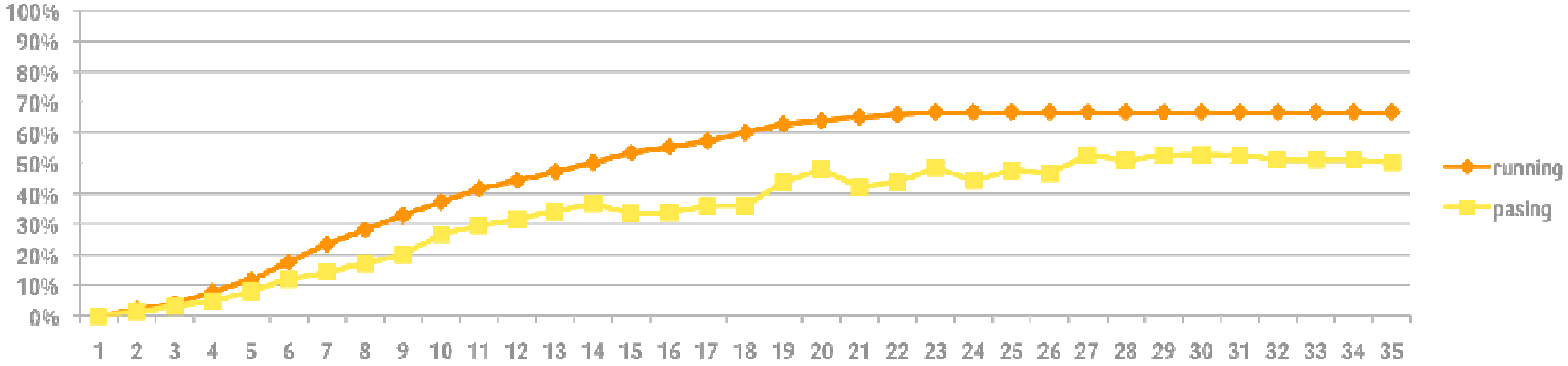
Keith Braithwaite

© Zühlke 2010

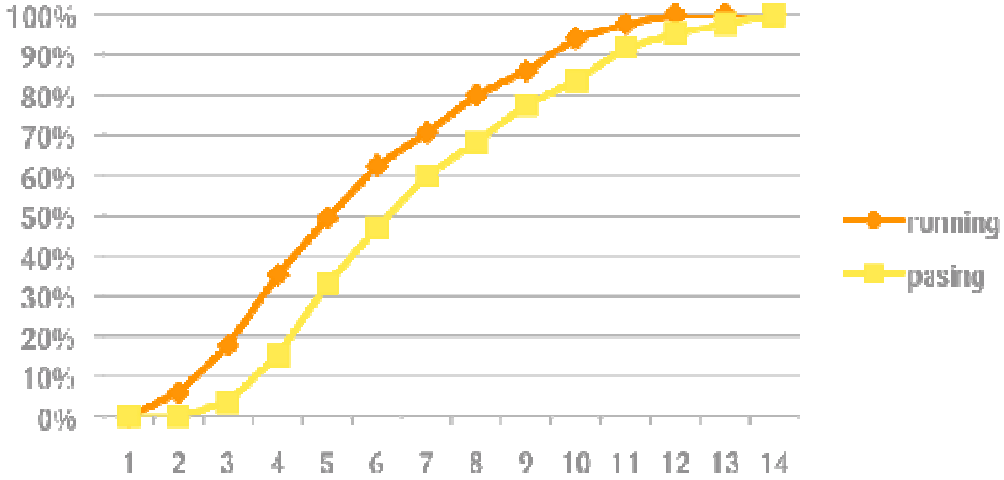
Hyperproductivity?

zühlke
empowering ideas

- Instead of this:



- This:



Keith Braithwaite

© Zühlke 2010

zühlke
empowering ideas

City and Guilds COBOL

- 3 attempts to compile, run and test or fail

There was a time when this sort of thing made sense

Keith Braithwaite
© Zühlke 2010

zühlke
empowering ideas

There was a time when this sort of thing made sense

Invention, Innovation and Diffusion of Software Development Techniques

Keith Braithwaite
© Zühlke 2010

zühlke
empowering ideas

Jerry Weinberg tells of being told that

- The computer (singular) earns more than you do, so behave accordingly

Keith Braithwaite
© Zühlke 2010

Econmics

The computer learns more than you, behave accordingly

- cost(processor time) >> cost(developer time)
- Cycle time to get feedback—hours to days

Keith Braithwaite
© Zühlke 2010

Economics

zühlke
empowering ideas

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.

Kent Beck
Mike Beedle
Arie van Bennekum

Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt

Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber

Jeff Sutherland
Dave Thomas

© 2001, the above authors
this declaration may be freely copied in any form,
but only in its entirety through this notice.

Diffusion?

zühlke
empowering ideas

NASA Project Mercury 1961–63

> Project Mercury was run with short half-day iterations. There was a technical review of all changes, and—interestingly—the Extreme Programming practice of test-first development was applied: Tests were planned and written in advance of each micro-increment, and then the code was written to pass the tests. Each mini-iteration required integration of all code and passing of tests.

"Agile and Iterative Development: a Manager's Guide"–Larman, 2004

Keith Braithwaite
© Zühlke 2010

Technology Changes Econmics

The computer learns more than you, behave accordingly

- cost(processor time) >> cost(developer time)
- Cycle time to get feedback—hours to days

In fact, you earn much more than the computer

© Zühlke 2010

Econmics

zühlke
empowering ideas

The computer learns more than you, behave accordingly

- cost(processor time) >> cost(developer time)
- Cycle time to get feedback—hours to days

You earn much more than the computer, behave accordingly

- cost(processor time) << cost(developer time)
- Cycle time to get feedback—milliseconds to minutes
- A top-end dev workstation amortised over 3 years
  - £1 per day
  - 2 or 3 *orders of magnitude* cheaper than a programmer

© Zühlke 2010

What next?

zühlke
empowering ideas

Characteristics of current landscape

- Processor cycles are near zero cost
- Memory is abundant
  - But working sets still might not fit
- Programmers are more abundant than ever
  - But good ones are still vanishingly rare

Keith Braithwaite
© Zühlke 2010

Diffusion from Research?

- Functional programming
- Proof assistants
- Model checking

Keith Braithwaite
© Zühlke 2010

Diffusion from Industry?

- Massive parallelism on massive datasets in the cloud
- Statistical algorithms
- Ubiquitous processing power

Keith Braithwaite
© Zühlke 2010

But still…

zühlke
empowering ideas

People write software for people with people

- Individuals and their interactions trump processes and tools
- Every time

Keith Braithwaite
© Zühlke 2010