



How do we achieve more with less?

Why Agile /
SCRUM

HaMIS story

Prerequisites



Edwin de Werk: @eworx4me en Viktor Grgić: @vrgric

Achievements

Every 3 weeks we ship software in production

Improving Architecture

“Functionality” delivered within constraints time and €

Happy users

5 stars TÜVIT



Satisfied Business, stakeholders

Architect & project manager can get along :-)

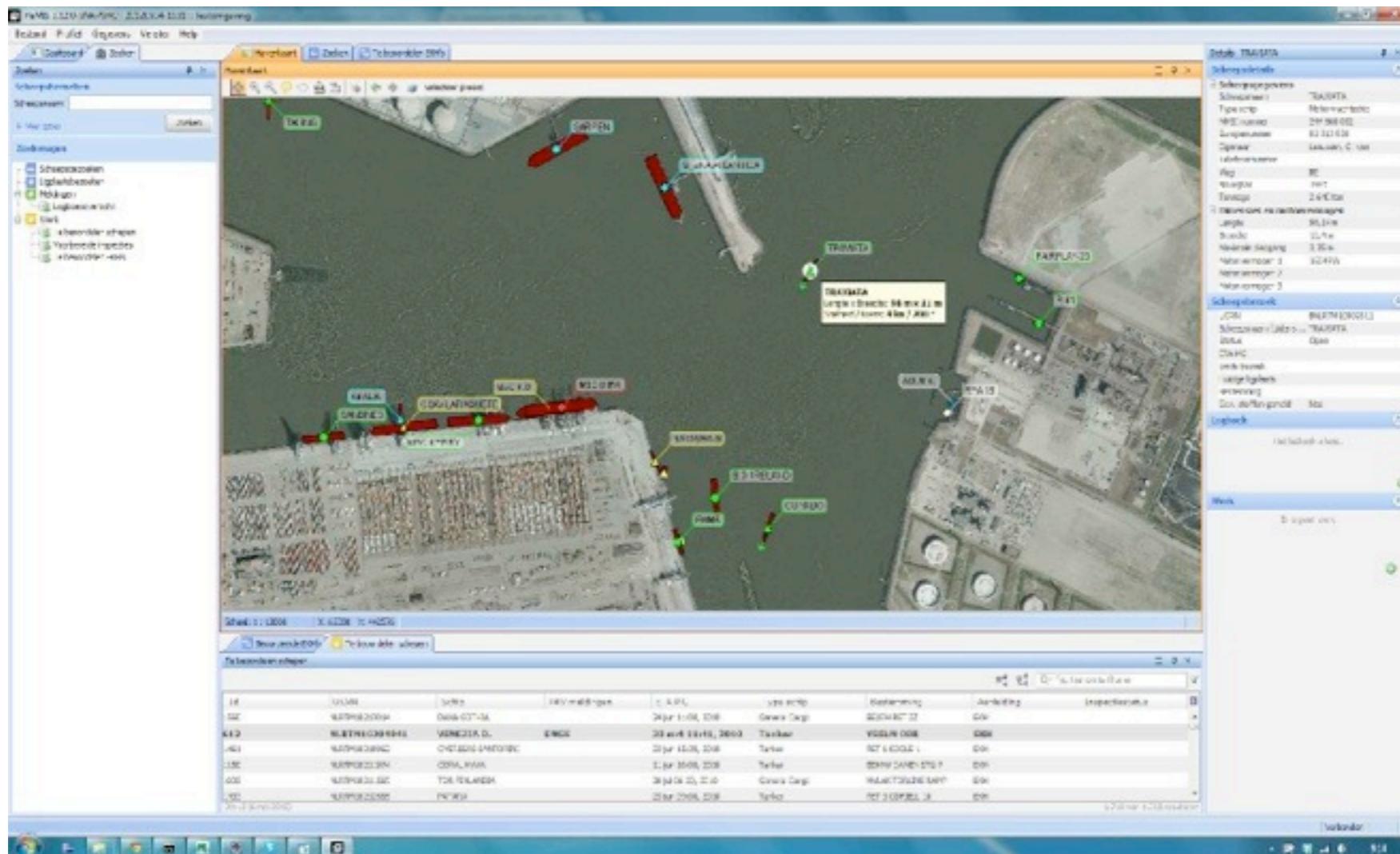
Developers like environment and challenges



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

HaMIS, Port of Rotterdam, Harbour master Management Information System.

24/7 application for traffic management, transport, environmental safety and crisis management.



Goals:

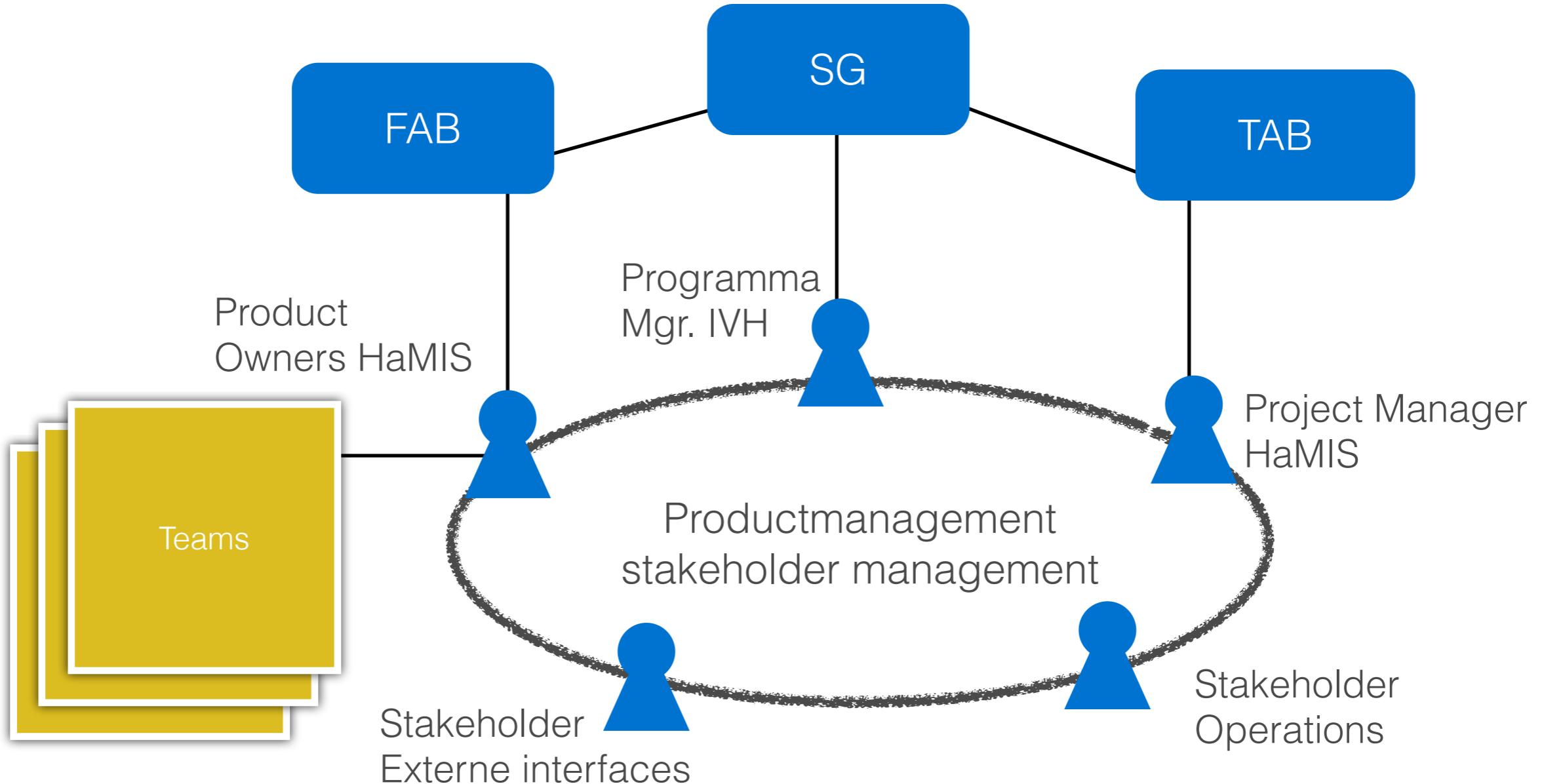
1. Replace outdated system
2. Support the port of Rotterdam expansion



Edwin de Werk: @eworx4me en Viktor Grgić: @vrgrgic

HaMIS,

Port of Rotterdam, Harbour master Management Information System.



Where do we come from?

100+
identified
use cases

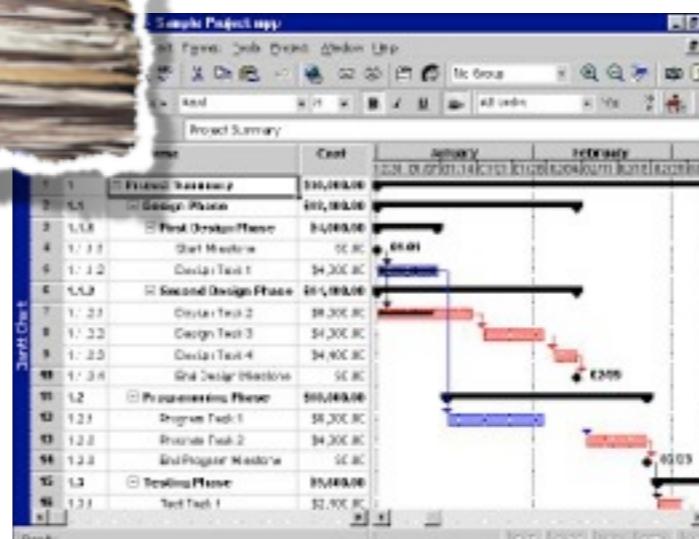
Project Start
Architecture



Q&A
department

BPEL process
server

RUP and Scrum but



Architecture with
4 layers & 6
subsystems

6 architects

800+ architectural
decisions



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

Resulting in...

One use case
in 6 months!



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

The assumption is...

In advance the

Customer knows what he wants!

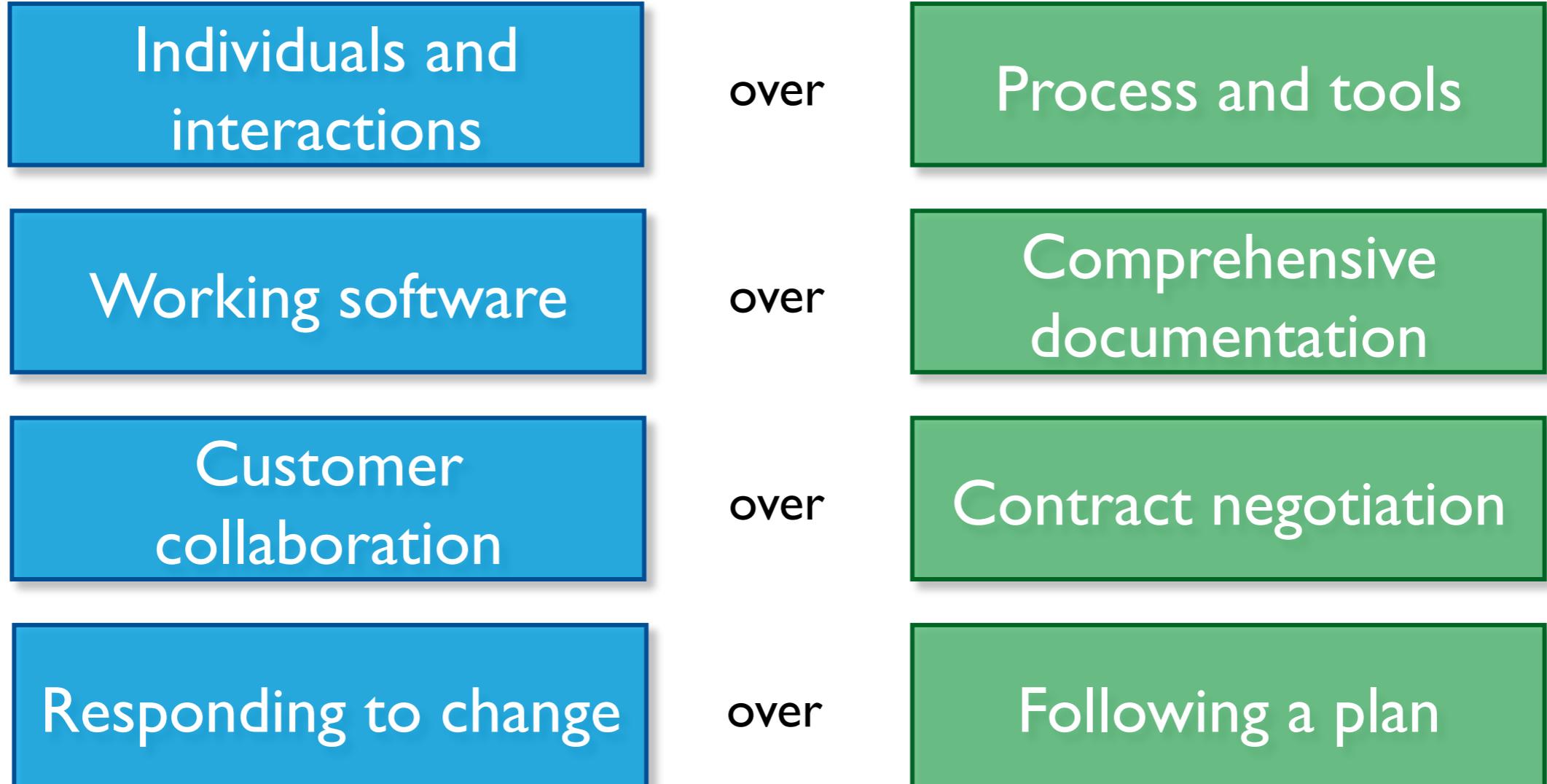
Developer knows how to build it!

And nothing will change !



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

Our Solution, Agile Manifesto



Source: www.agilemanifesto.org

The most important for success

Hire coaching

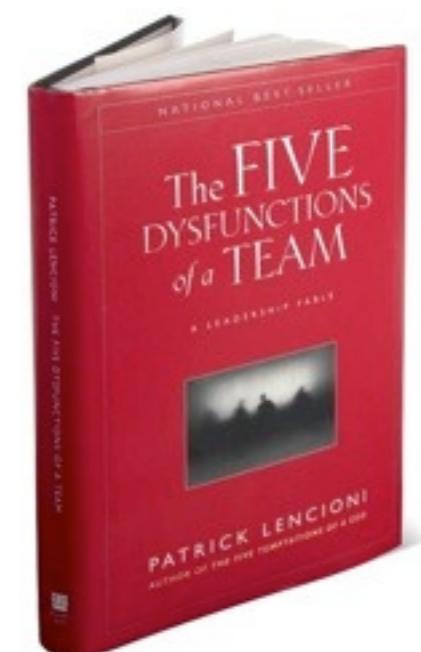
Work the system.



Right people, no sheldons



Team

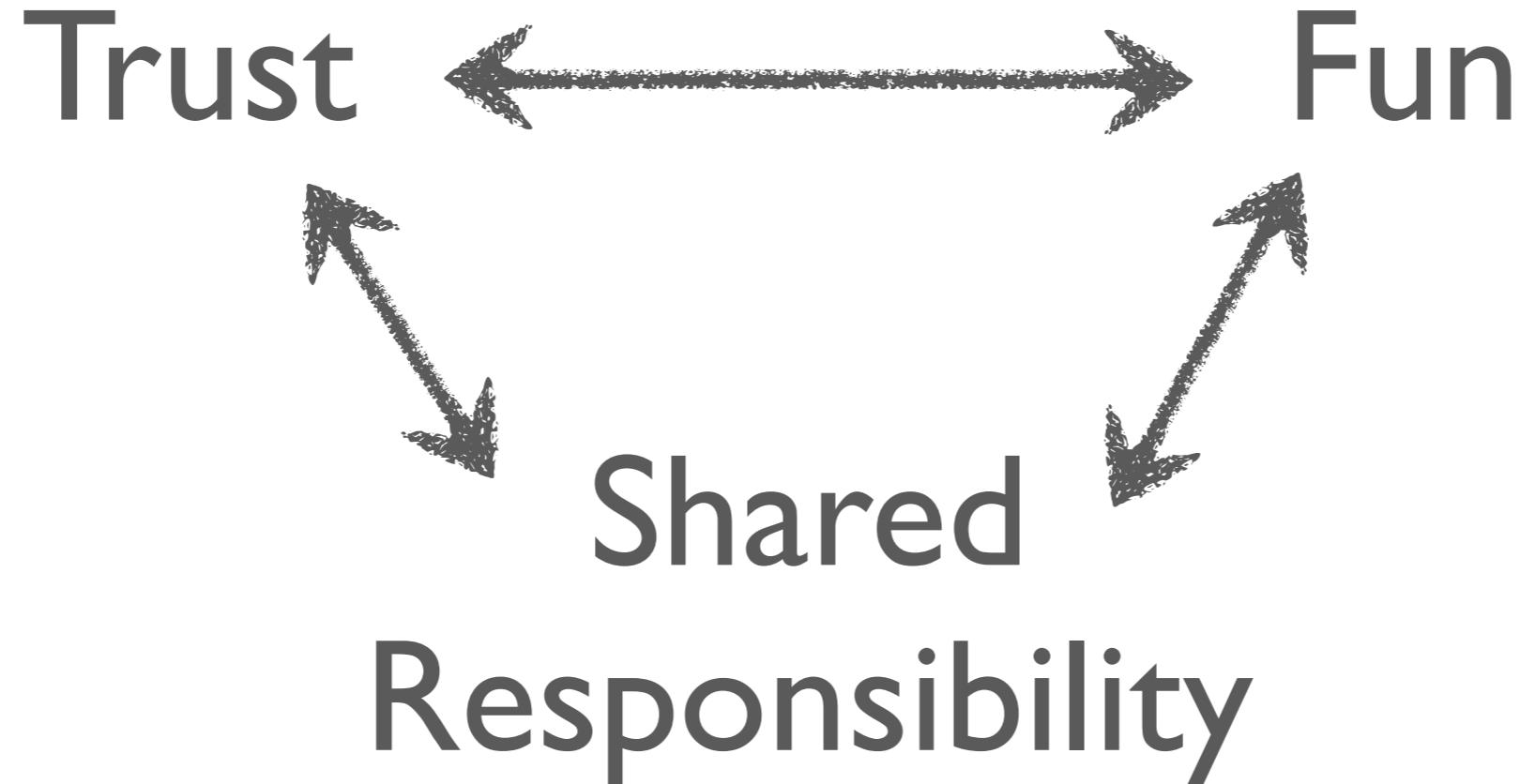


**Grow a pair and try,
learn, try**



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgc

The most important for success



Team

Trust before
decision
making

What am I
proud of?

Diner after
sprint

Workshops in
hotel

ski trips in
France



Quality / Architecture extremely important

Just-in-time
just-enough

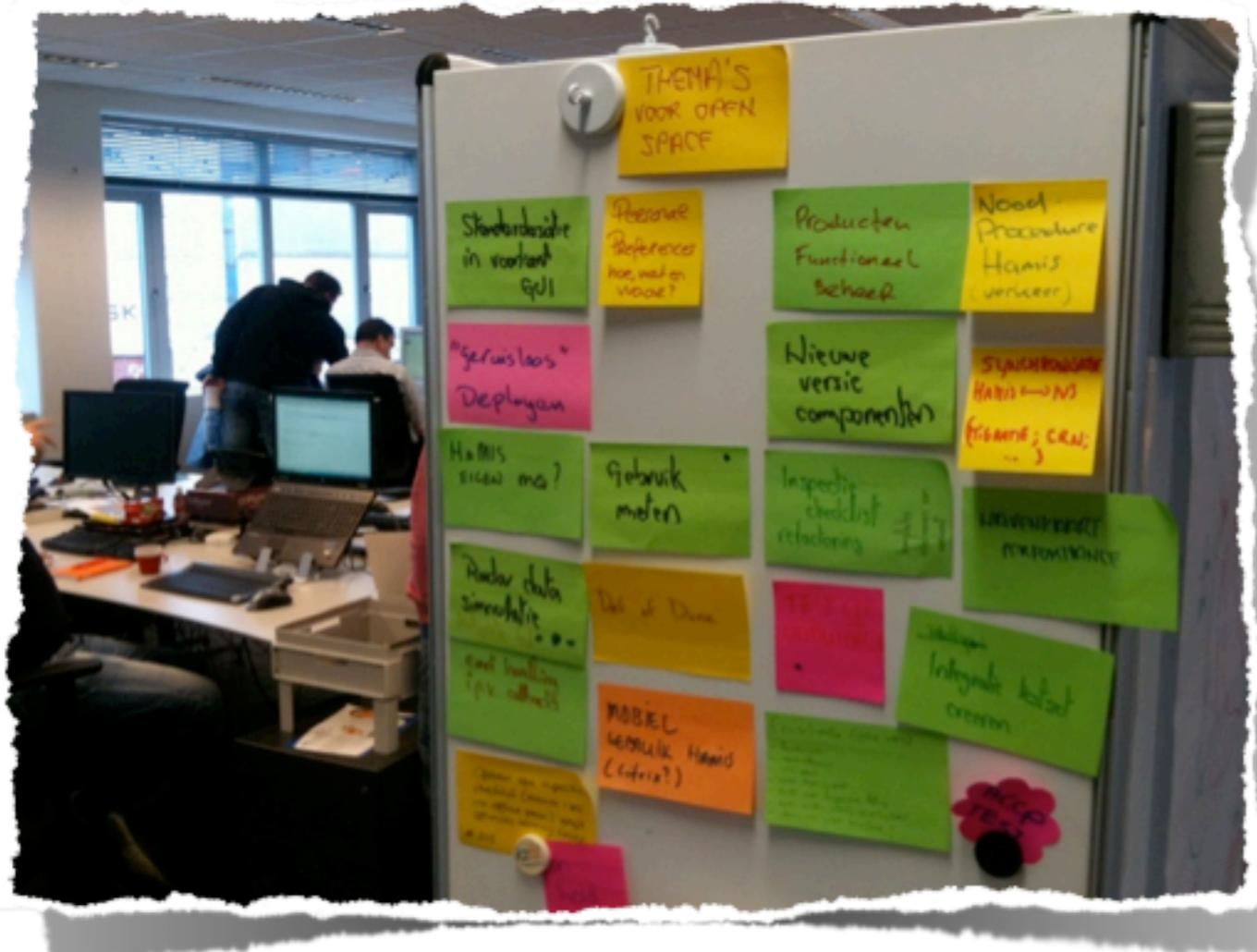
Open space
sessions

Knowledge
sharing

Documentation is Wiki
filled by “harvesting” the
knowledge

Standardization is
discovered and not
predefined

Continuous
refactoring towards
better design

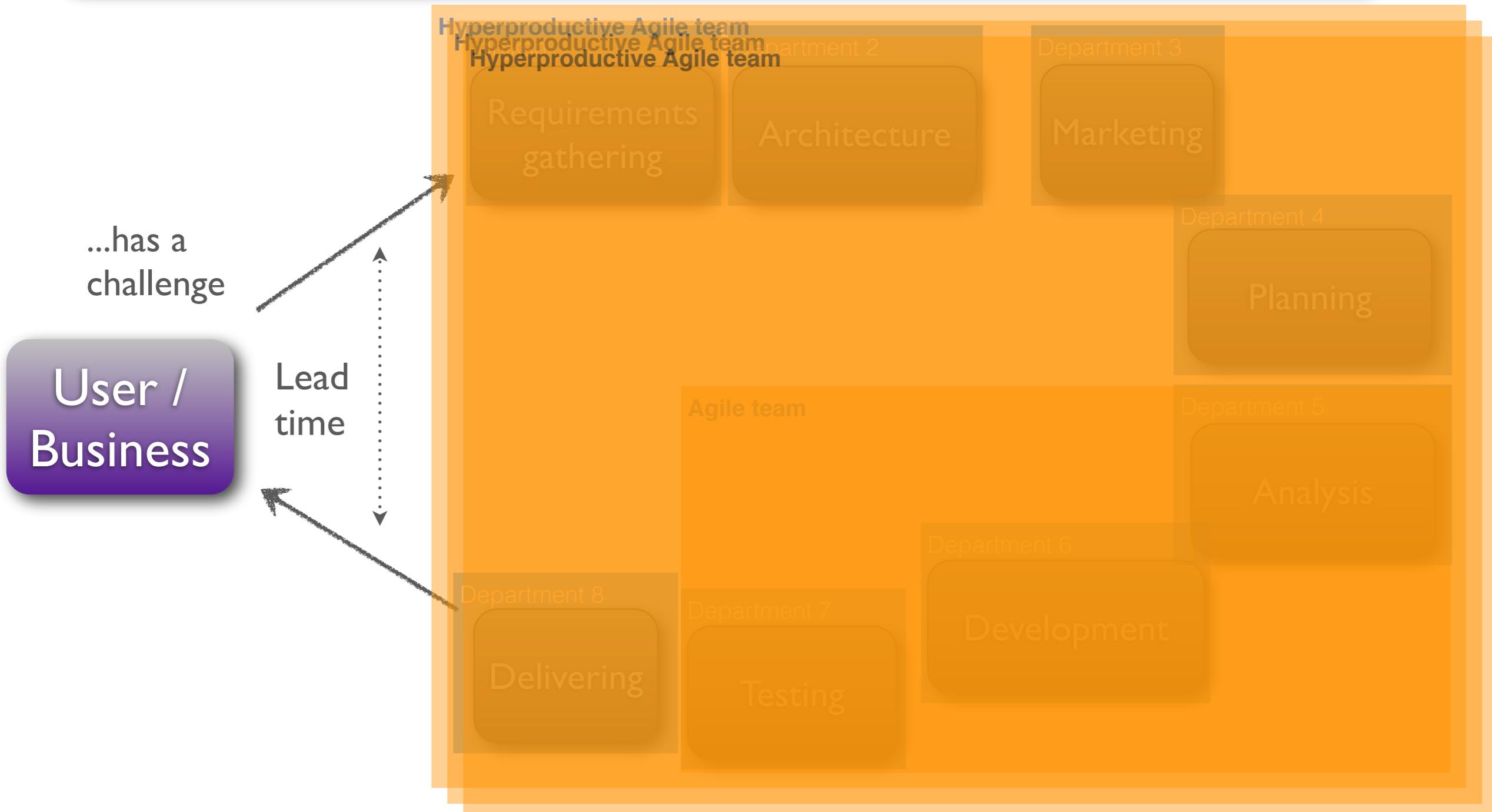


Architecture principles:
Simplicity and design for change



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

Optimize the whole

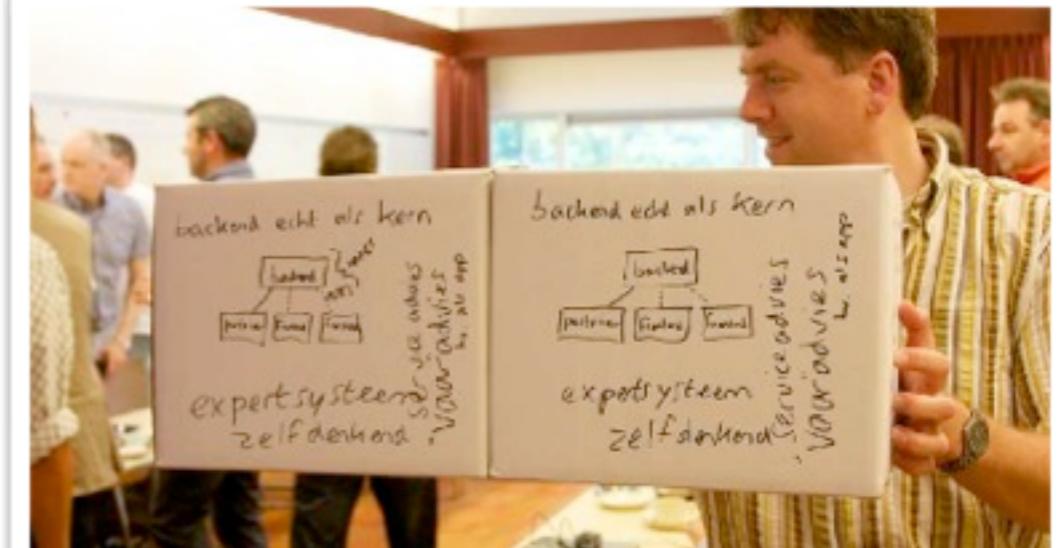


Pull, continuous flow, customer value, eliminate waste, continuous improvement



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

Product vision, creativity, commitment



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

Team members and user



No translation between the
customer and developer mindset!



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic



**“Everybody,
altogether,
early on”**

James O. Coplien

Challenges



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgcic

Work the system

Develop, protect and direct with constraints

Intrinsic motivation

Make surroundings Scrum friendly

Change advisory board vs. sprint release



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgcic

Beyond Scrum

We don't ask anymore
“What does Jeff say?”

Agile / Scrum is part
of DNA, we just do it!

Stories are more and
more defined by teams

Real production instead
of production ready



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgcic

Behaviour & Titles



“Bureaucracy is here to compensate incompetence and lack of discipline”

I am an
- architect
- manager
- designer
- analyst
- scrum master
...



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgic

Retrospectives

Games can become a goal

Not going “deep” enough



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgcic

Team or “virtual” team

Backlog
grooming

Which questions are solved by one of the teams and which together

Division of stories



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgcic

How did we achieve more with less?

Less...

- * Control
- * Whish lists
- * Procedures and processes
- * Architects, analysts, testers, team leads, project managers,... in traditional sense
- * Technology push, (off-the-shelf) products and tools with no direct value
- * Generic and just-in-case solutions
- * Beforehand documentation
- * Extensive use-case descriptions
- * Chickens!

More...

- * Concrete goals
- * Vision
- * Face-to-face customer collaboration
- * Team responsibility and decision making
- * Continuous improvement
- * Trust
- * Demand pull
- * Feedback loops
- * Solutions for harbor master instead of “future-proof” and “flexible statements”
- * Pigs!



Edwin de Werk: @eworx4me en Viktor Grgić: @vgrgcic

Meer weten?



Edwin de Werk: @eworx4me

Viktor Grgić: @vrgric