



Agile @ Siteimprove

Mikkel Toudal Kristiansen, Agile Coach at Siteimprove

Scrum Day Denmark 2018

About me

Mikkel Toudal Kristiansen
@otwtbs
mikkel.kristiansen@gmail.com

1995

First job as a
developer

2005

Agile, Scrum,
Kanban and
Lean evangelist

2013

Professional
Scrum Trainer
with Scrum.org

2018

Agile Coach at
Siteimprove



Danish SaaS company founded in 2003

100 employees in 2014 – 530 today!

Offices in 9 countries – HQ is still in the heart of Copenhagen

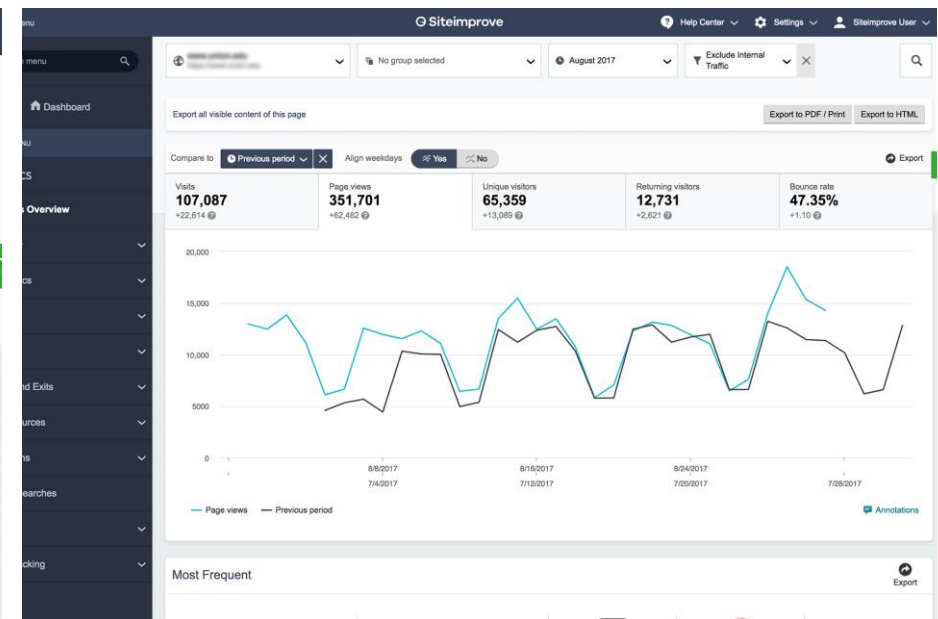
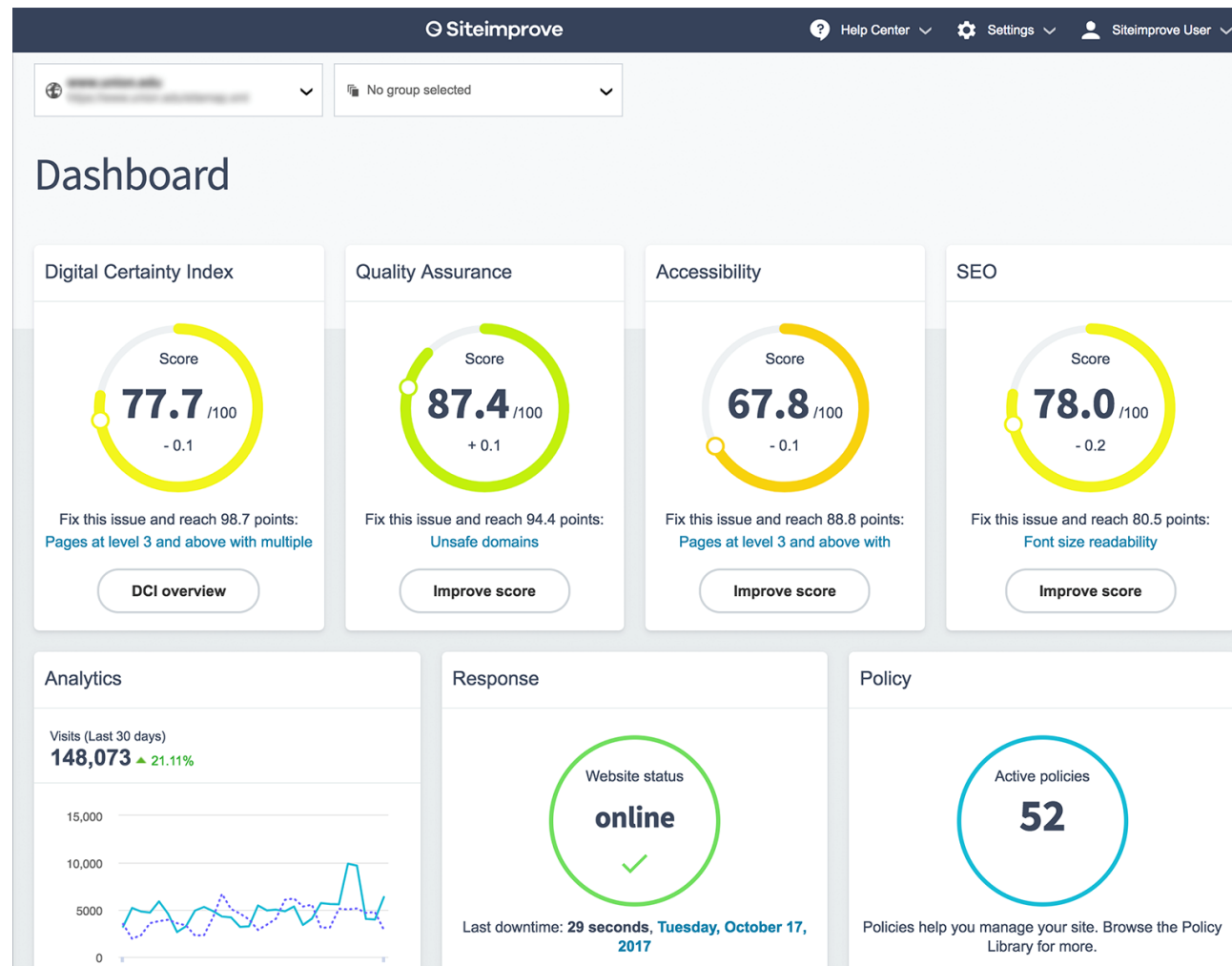
All development happens at HQ

Began as a broken links checker

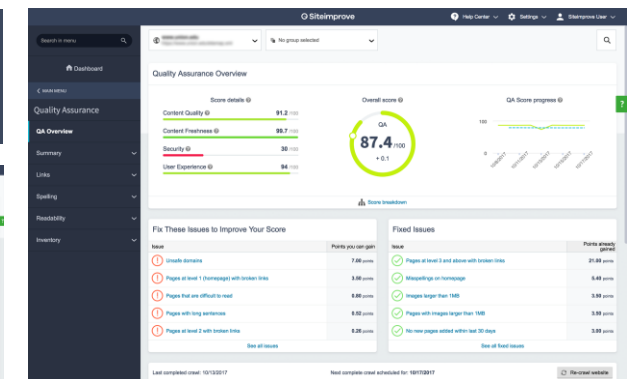
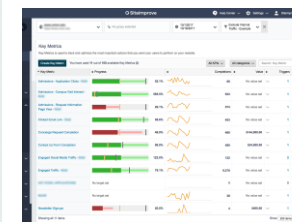
Now also checks accessibility, GDPR compliance, SEO, and much more ...



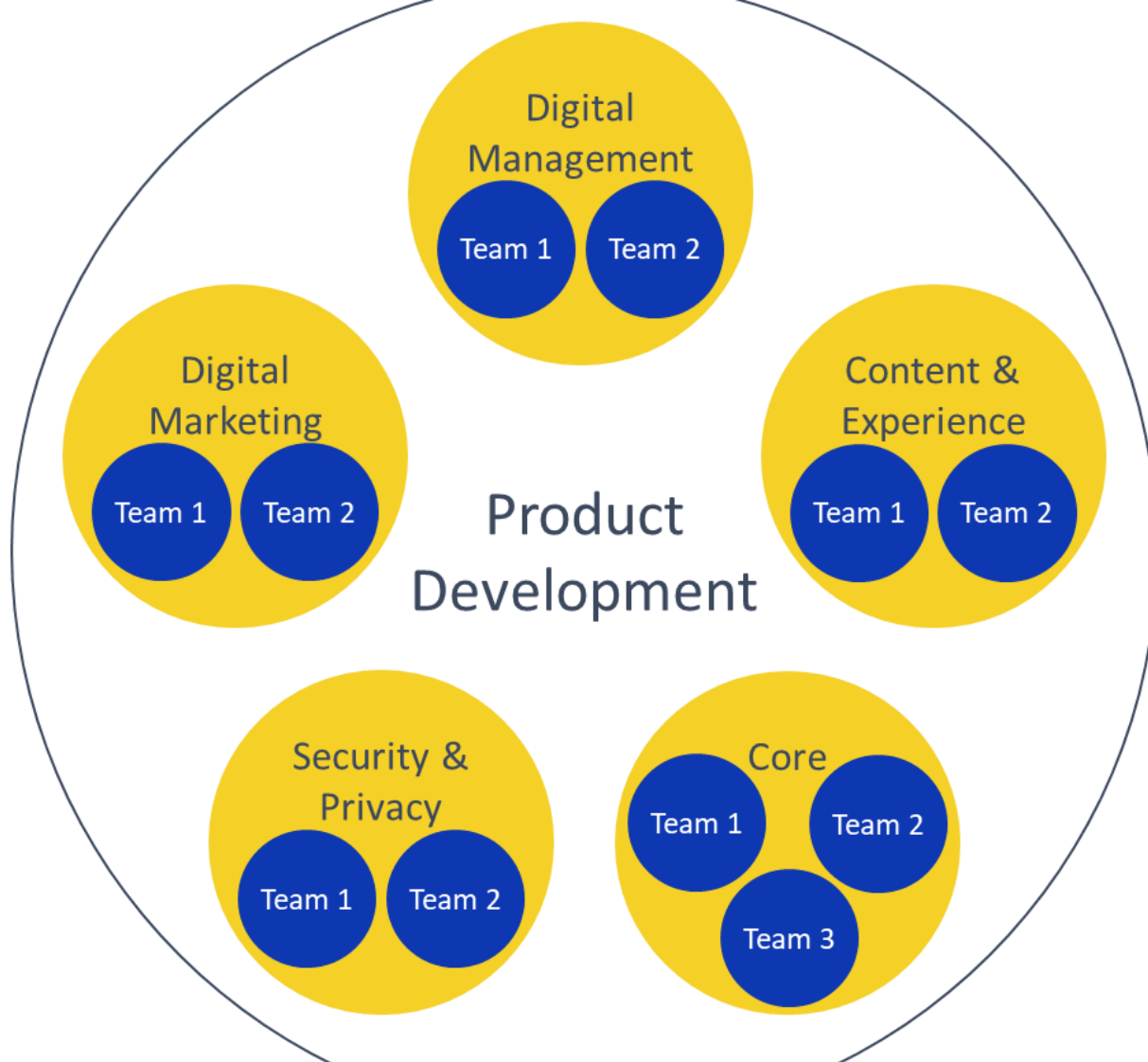
 **Siteimprove**



And much more ...



Our product



The product is split into several functional areas, so ...

We are organized into 4 Product Units and a Core Unit

A Unit is split into 2-3 Teams

Organization

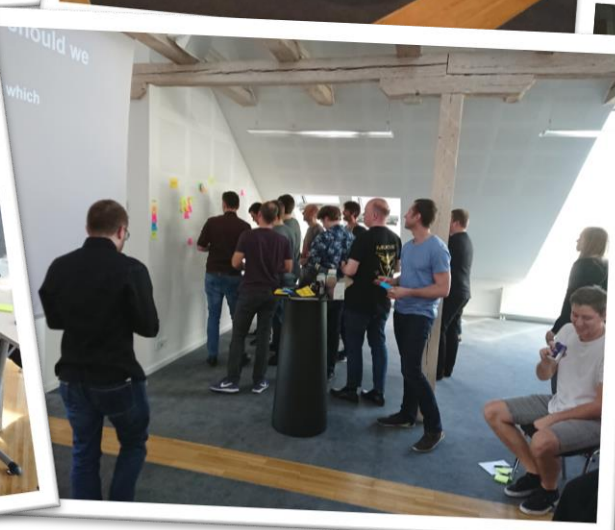
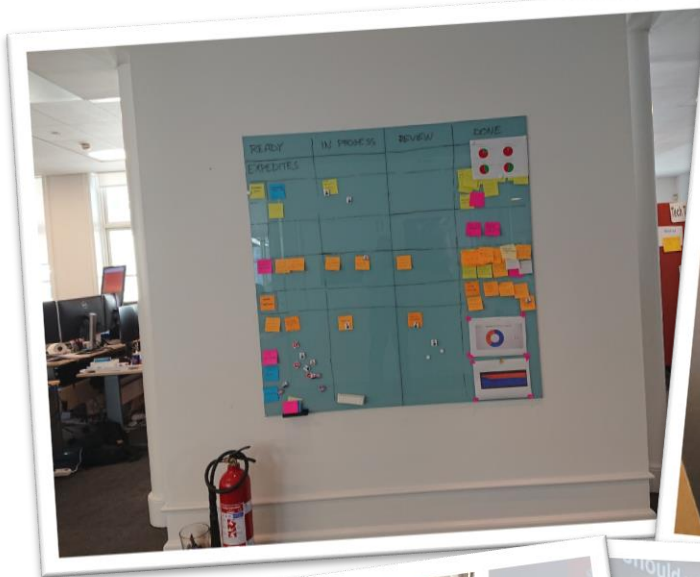


Teams at Site improve

- are "permanent"
- are co-located at HQ
- are cross-functional
- use Scrum, Kanban, or a mix of the two

Collaboration in Teams

How do we work?





Every Product Unit has one or more UX specialists

Product Owners and UXers ideate and design prototypes

We run Alpha and Beta tests with FirstImprovers

We want to include customers more in our design process

- Design Sprints
- Co-creation
- Early and frequent validation of assumptions

UX and Agile?



6 months – let's evaluate!

The good ...

Product Units and Teams are major improvements!

Cross-functional Teams work!

Lots of new features and functional areas released!

The bad ...

However, coordination, alignment and collaboration are challenged

Across Teams in a Unit

Across Units

With the rest of the organization

We have become too siloed

6 months later ...

More management



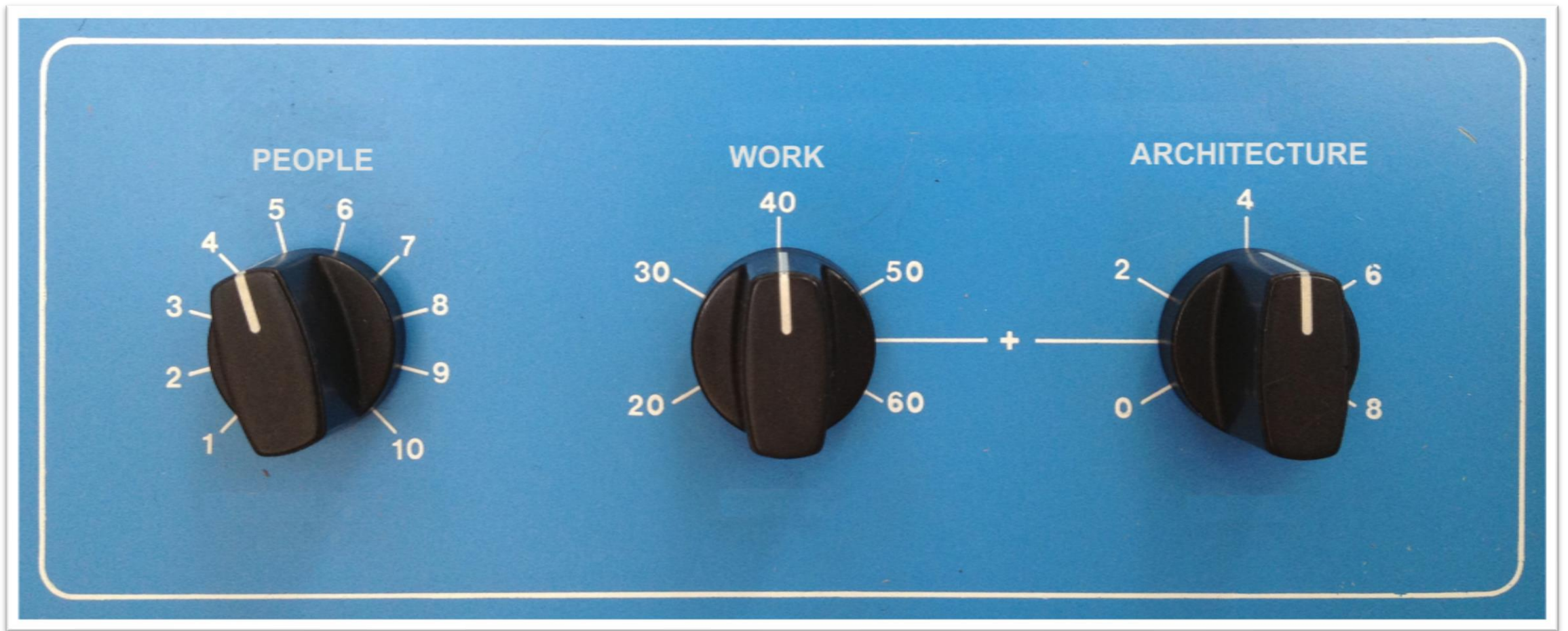
More meetings

More documentation

Stricter processes for handovers

Less productivity

The traditional solution

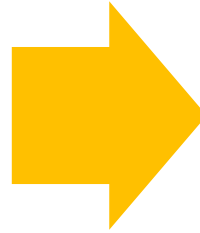


The available levers

Establish a loosely coupled architecture

One core platform

A number of plugins, each depending only on the core



Benefits include:

Plugins can be owned by single Units or Teams

Less coordination is necessary

More autonomy for Units and Teams

Options – architecture

Make the
Product Units
and Teams even
more cross-
functional

Introduce
scaling practices
from the Nexus
framework

Options – people

Organize work to happen in parallel instead of sequence

- Release management
- Marketing materials and campaigns
- Training of Sales and Support

Manage dependencies explicitly and deliberately

- Between people, Teams, Units and beyond
- Between items on the Product Backlog
- On software components and technology

Options – work

Currently, one single Nexus-like structure is sufficient

As we grow, we may need multiple Nexuses

The architecture (core + plugins) will enable several paths

Each Product Unit could become a Nexus

Or we could group 2 or more Product Units into a Nexus

But what about the future?



Thank you!