

# Taming Agile Architecture





We solve  
hard IT problems.

## Facts and Figures



Foundation in 2001



Located in Unterföhring  
near Munich



99% University Degree Holder  
60% with a PhD



2 Internal Training Days per Month



Approx. 275 Employees



41% with a Degree in  
Physics

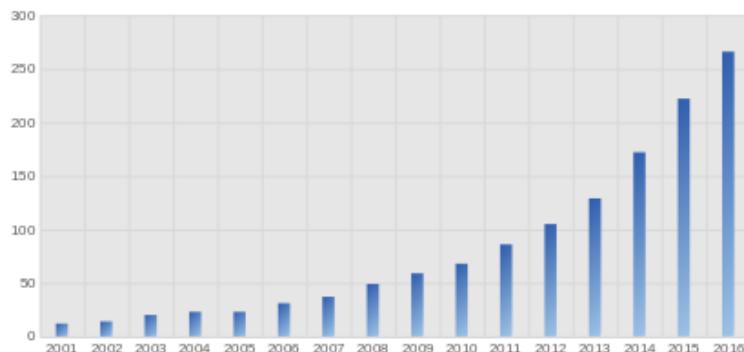


Fluctuation < 6%



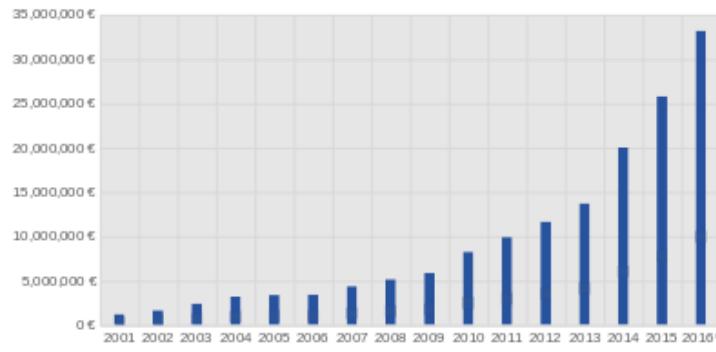
1-4 External Conferences per Year

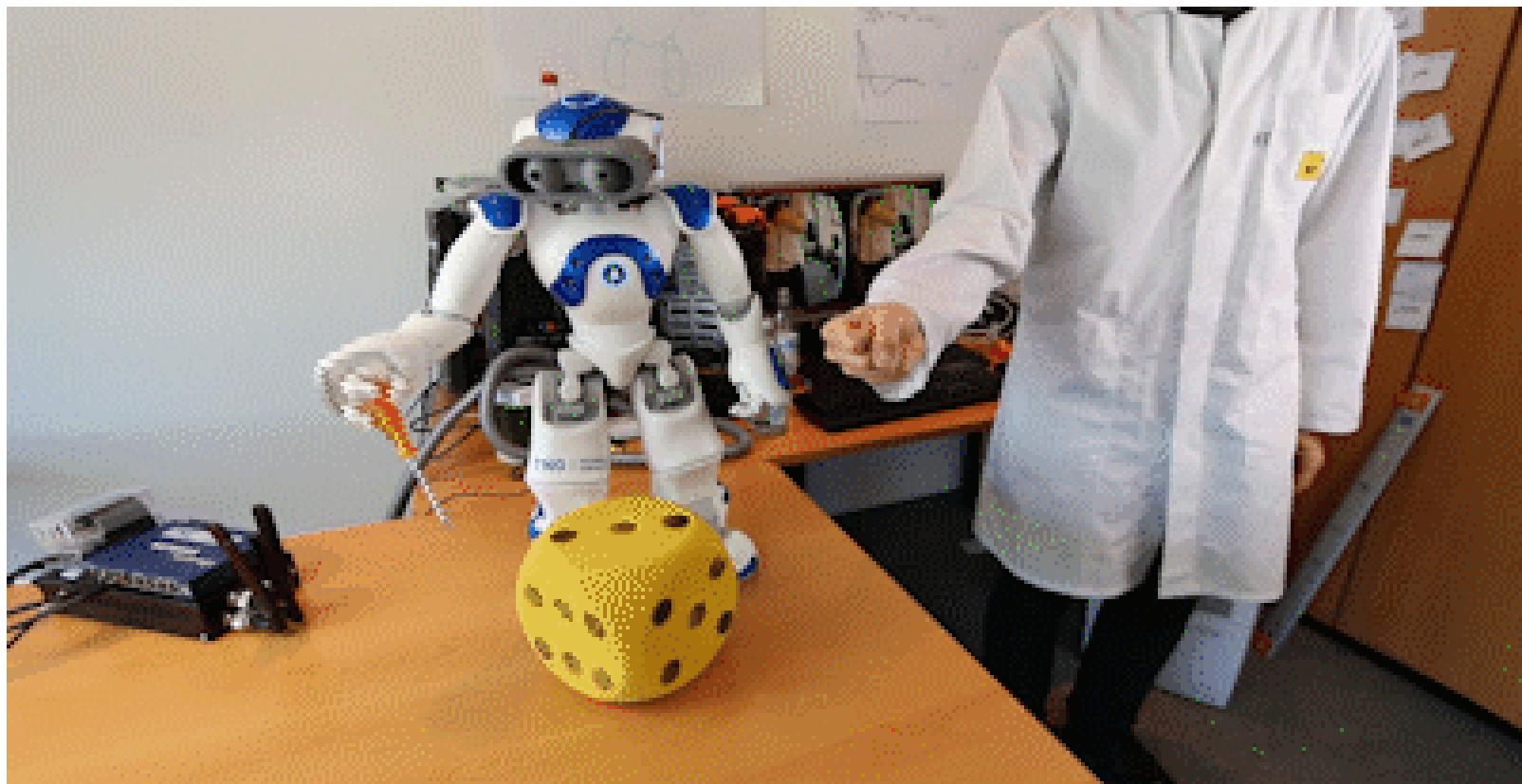
Number of Employees



Note: The number of employees is stated as of the end of December.

Sales Growth







# Disclaimer

*The opinions expressed in this presentation and on the following slides are solely those of the presenter and not necessarily those of TNG Technology Consulting.*

A photograph showing a group of approximately ten people in a workshop or office environment. They are standing around a large, round, light-colored wooden table. Some individuals are leaning in, looking down at the table, while others stand back, engaged in conversation. The room has warm lighting and appears to be a creative or professional workspace.

*"Individuals and interactions over  
processes and tools..."*

- Agile Manifesto

**Let's be agile, lets interact!**

# Agenda

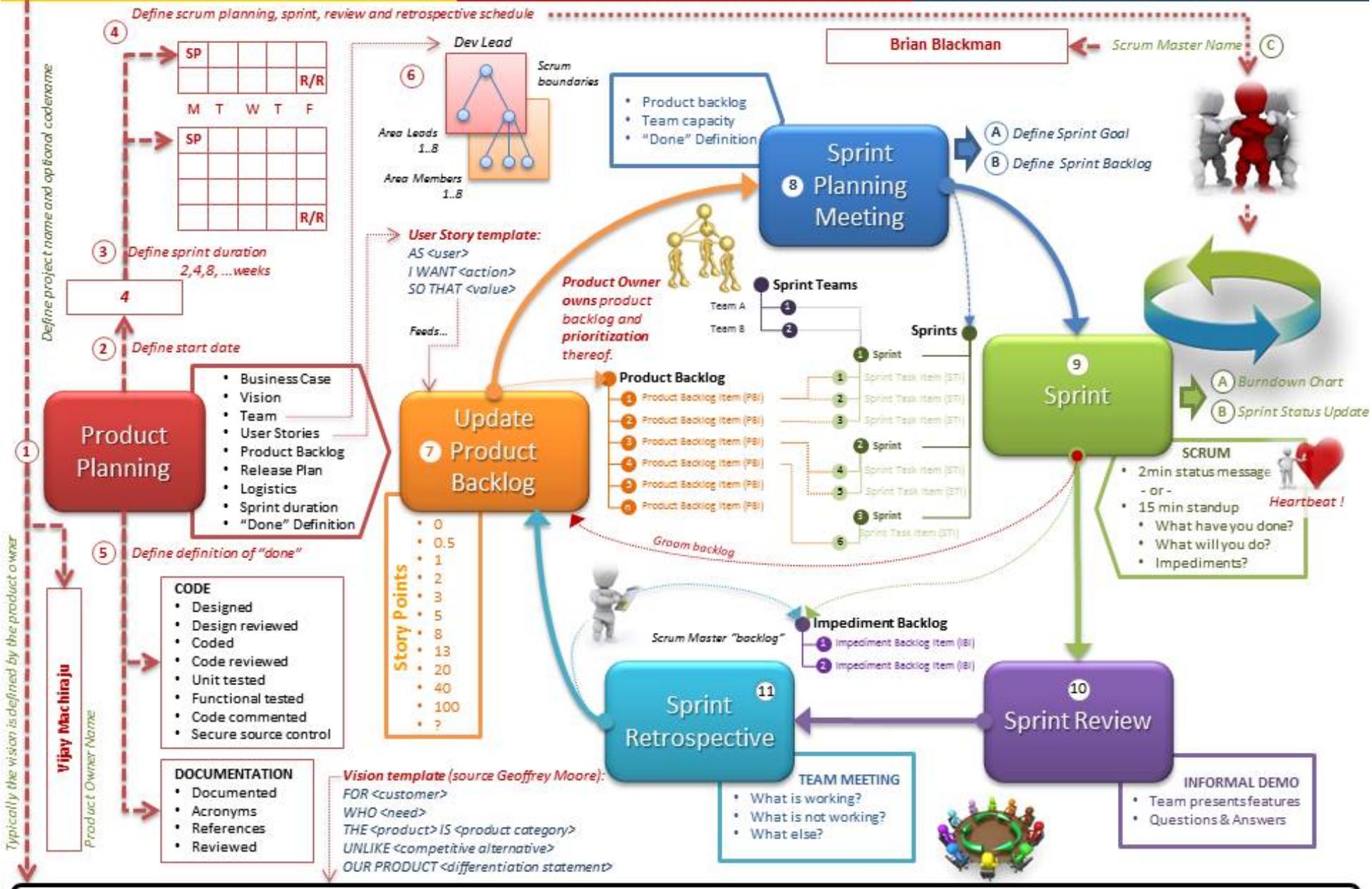
Agile/Architecture Myths

Practicing Agile Architecture

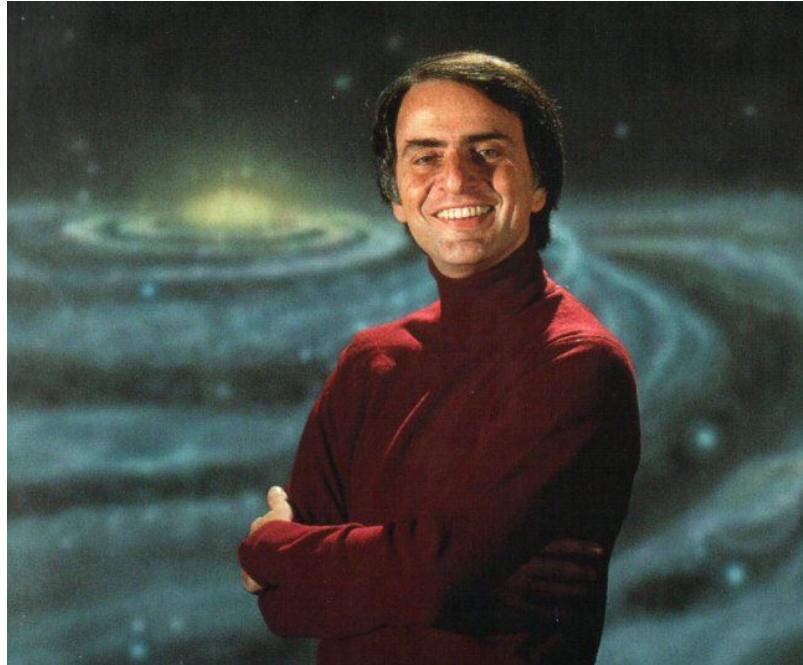
# Why do we even talk about this?

*“The word 'agile' has been subverted to the point where it is effectively meaningless, and what passes for an agile community seems to be largely an arena for consultants and vendors to hawk services and products.”*

- *Dave Thomas*



# The Fine Art of Baloney Detection

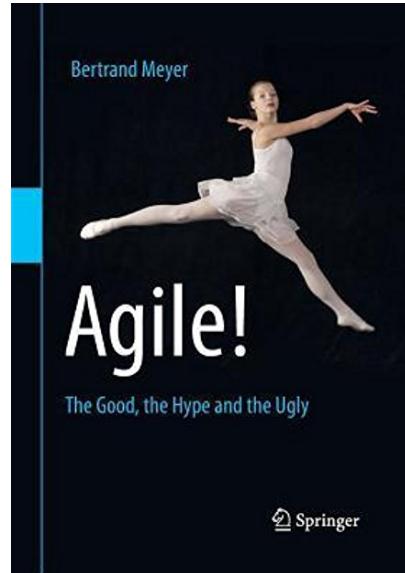


*"The Demon-Haunted World: Science as a Candle in the Dark"* - Carl Sagan

- there must be independent confirmation of "facts"
- arguments from authorities carry little weight
- try not to get overly attached to a hypothesis, just because it's yours
- ...

# Agile!

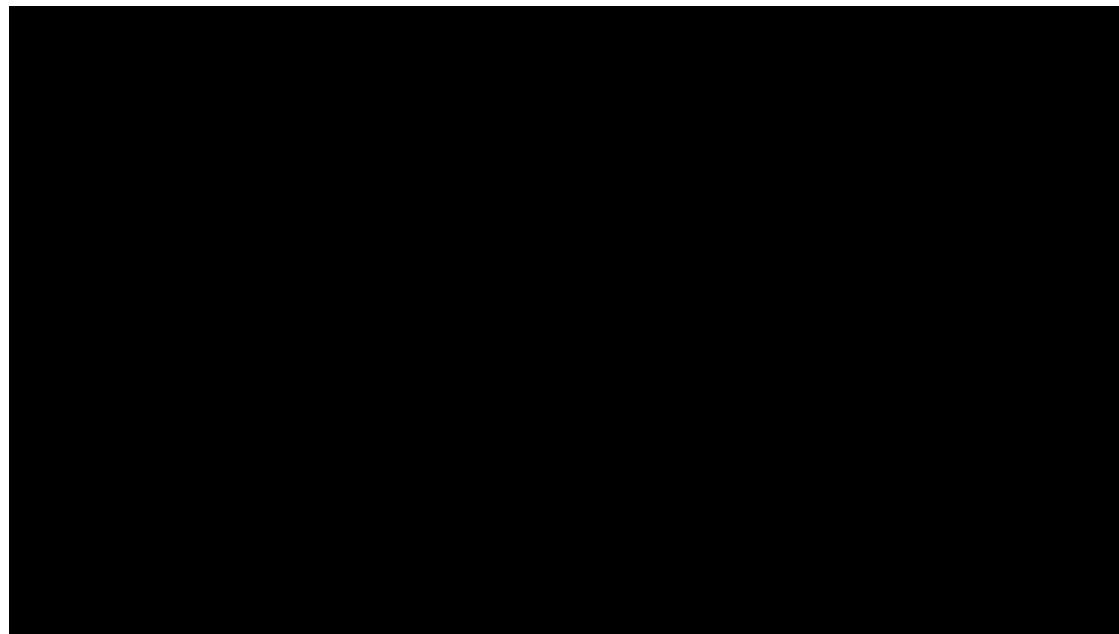
## The Good, the Hype and the Ugly



Bertrand Meyer

# ”One Hacker Way”

by Erik Meijer



# Agile Manifesto

*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.**

# Myth #1

Agile means no documentation





## Myth #2

### Agile means no architecture



# Myth #3

## Good coding practices lead to good architecture

A close-up photograph of a baby lying on its back on a light-colored surface. A woman's hands are visible, gently holding the baby's legs. The baby has a pacifier in its mouth and is looking towards the camera. In the background, there is a blue wooden toy box.

## Myth #4

Agile means everything is easy to  
change  
(even architecture)



**Myth #5**

**Scrum === Agile**

# Myth #6

We're agile because ...

- ... we're using Jira
- ... we're doing Standups
- ... we use microservices
- ... we're using DDD

# SAY A GLASS



## ONE MORE

# TIME

# Agile Architecture



# Architecture

*"Software architecture refers to the high level structures of a software system, the discipline of creating such structures, and the documentation of these structures."*

*- Wikipedia*

# Architecture

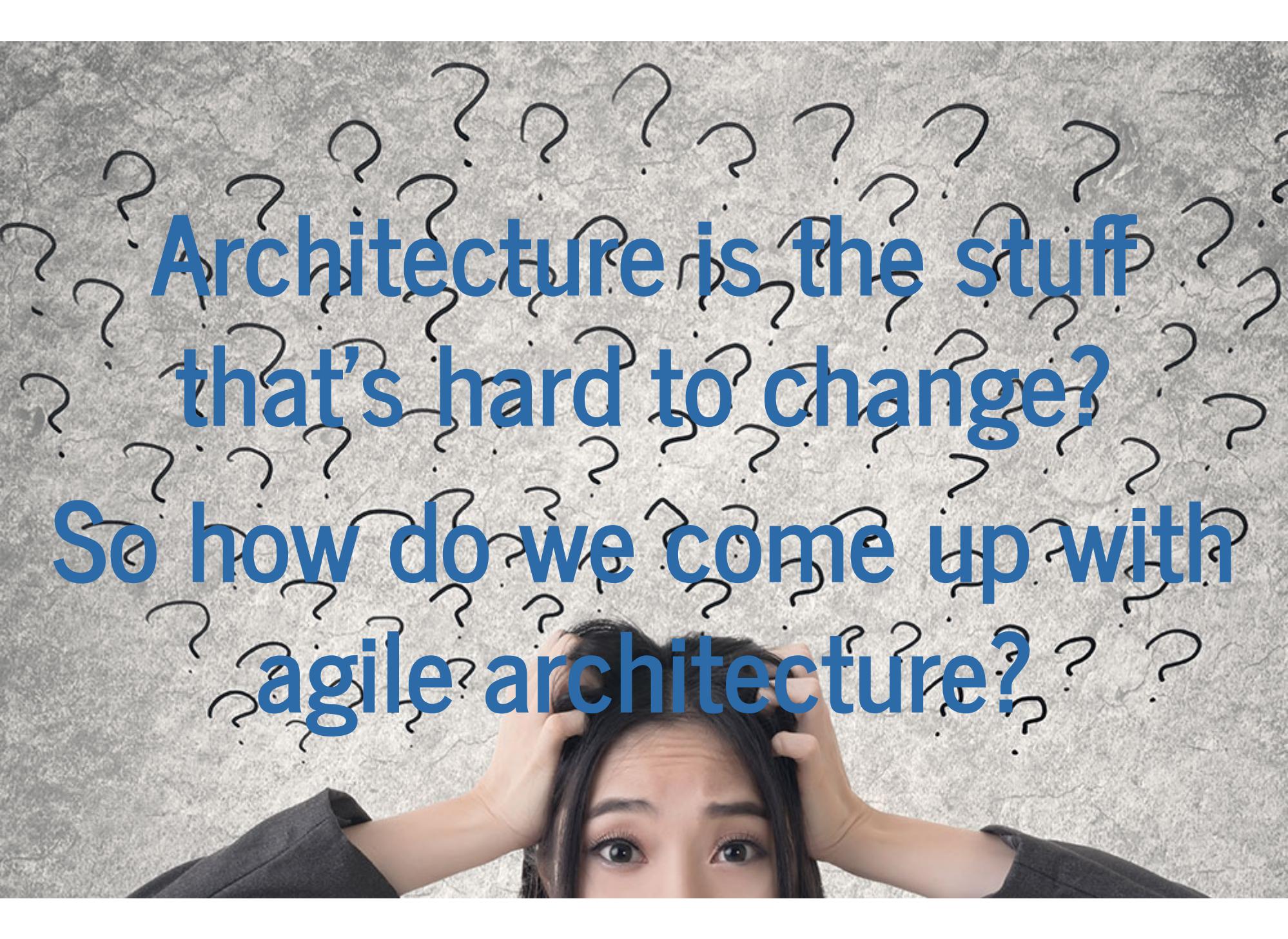
*"[Architecture] is the stuff that's hard to change"*

*"Architecture is not all about diagrams; it's about promoting collaboration and communication of the critical parts of the system."*

*- Martin Fowler*

# Problems with traditional architecture

- ivory-tower architecture
- making unnecessary decisions too early
- architect not part of the team
- architect not accountable



**Architecture is the stuff  
that's hard to change?**

**So how do we come up with  
agile architecture?**

# Gather all known requirements

Download the App	Register Person	Register an Account	Authenticate	Provide the Address	Select Payment Method	Place the Order
As a iPhone user I want to download app so I can order coffee	As a customer I want to collect & store customers details so that I can place orders easily	As coffee mob I want to collect & store customers details so that account details so that I can place orders easily	As a customer I want to login to my account so I can view the app with my preferences	As a customer I want to see drinks so I can pick drinks	As a customer I want to select cash or cashless type so I can pay	As a customer I want to order my coffee via my phone so that it saves me time
As an android user I want to download app in order to order coffee	As coffee mob I want to collect & store details so that I can receive orders	As coffee mob I want to collect & store details so that I can receive orders	As a customer I want to verify phone number so I can strongly authenticate	As the owner I want to update menu so I can be up to date	As a customer I want to save my address to pay my order with	As a customer I want to save my address to pay my order with
As coffee mob I want to offer the app free so more customers download it	As a customer I want to change my habitual order so I can add/change my order	I want to verify phone number so I can strongly authenticate	As the owner I want to manage price list so I can price drinks	As a customer I want to change my habitual order so I can add/change my order	As a customer I want to change my habitual order so I can add/change my order	As a customer I want to change my habitual order so I can add/change my order
As coffee mob I want to utilise our brand image/logo on the app so that it provides a link to our app from our website so customers can easily download	As a customer I want to use a credit card account so I can pay for my coffee with a credit card					



KEEP  
CALM  
AND  
PRIORITIZE

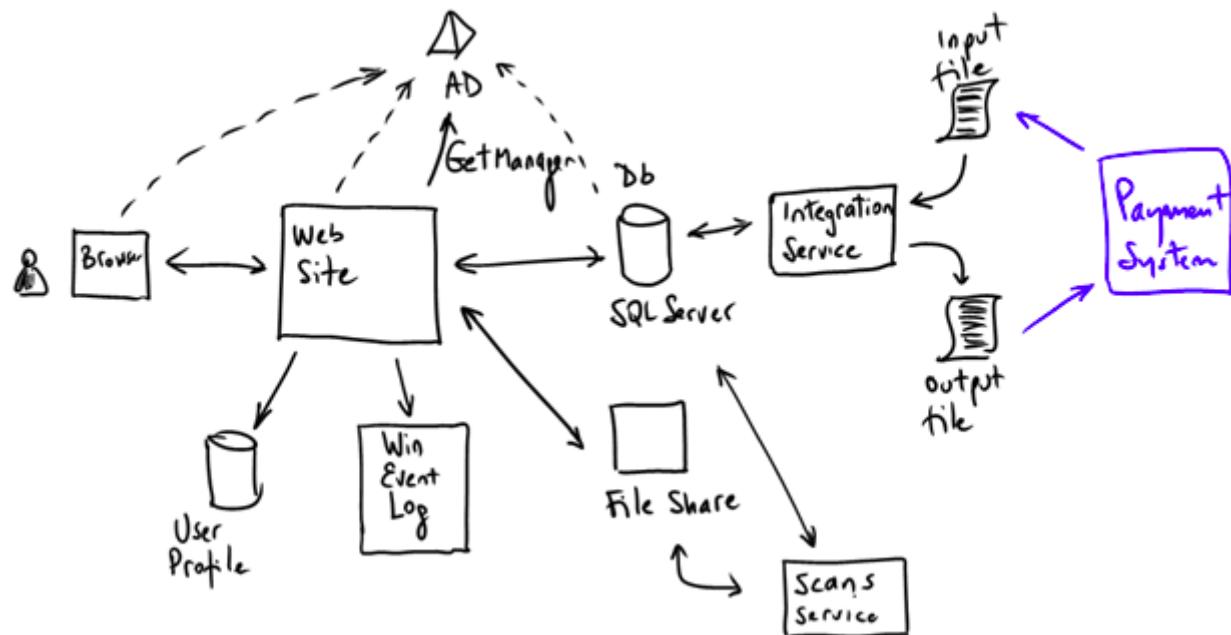
# Identify nouns and verbs

As a Game Player,

I want my Rocket to move back  
and forth when I press left and  
right arrows

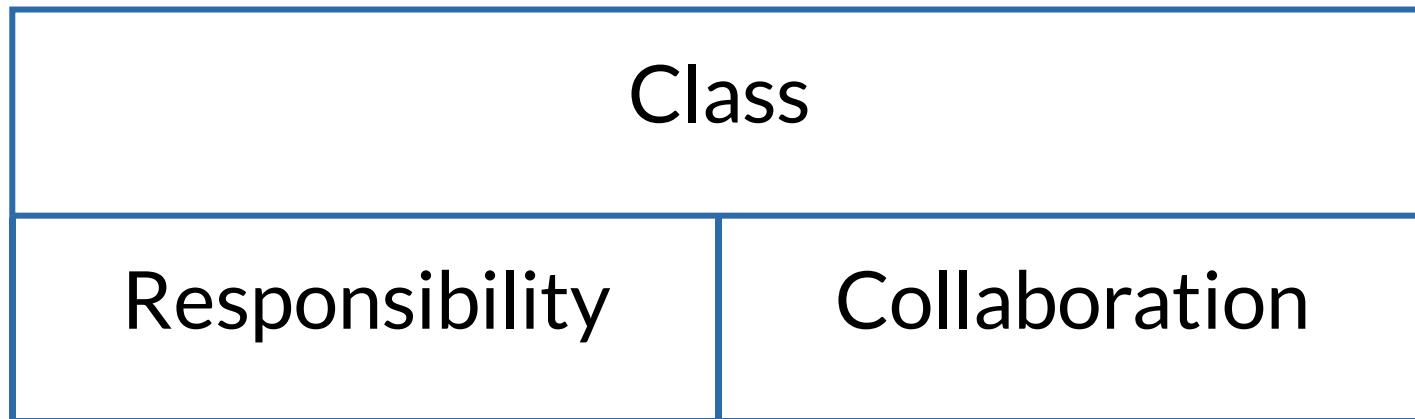
so that I can avoid asteroids

# Whiteboard Architecture





# Class-Responsibility-Collaboration Cards (CRC)



Lather  
Rinse  
Repeat

*"Factor your system so that artifacts that change at similar rates are together."*

- Foote & Yoder

# Test your architecture



# Go crazy

- native support for Arabic users
- Bitcoin check-out
- record user interaction for audits
- make all user actions undoable
- create native apps for iOS, Android, ...
- add offline support

# Design Decisions

- as little as possible, as much as needed
- think big, act small
- KISS - Keep It Simple, Stupid
- YAGNIY - You ain't gonna need it (yet)

# Architectural Decisions

*"Beware of people who love to think about abstractions more than coding software." - Steve Green*



# Microservices



**Daniel Bryant**

@danielbryantuk



Follow

"Beware of the distributed monolith connected via a database"

[@lornajane #CraftConf](#) meetup

6:06 PM - 26 Apr 2017



248

315

# Fallacies of distributed computing

1. The network is reliable.
2. Latency is zero.
3. Bandwidth is infinite.
4. The network is secure.
5. Topology doesn't change.
6. There is one administrator.
7. Transport cost is zero.
8. The network is homogeneous.

You are not Netflix.

You're probably fine starting  
out with a Monolith.

Think in modules/services

but

distribute over the network  
only if absolutely necessary

# Domain Driven Design

vs.

# Domain Centered

# Domain-Driven Design



# Domain Driven Design

- Tackling **Complexity** in the Heart of Software
- introduces overhead
- requires knowledge of existing architectural patterns
- not a one-size-fits-all approach

Same thing goes for Event Sourcing

# Everyday Architecture

- sprint 0, -1, ... for big projects
- architecture backlog items
- code reviews also from a 30.000 foot view
- retrospectives, ...
- in small/medium projects, everyone is the architect

*"An architect's value is inversely proportional to the number of decisions she/he makes."*  
- Martin Fowler

Bring your Architecture and  
Documentation back to life

# Revive your Documentation

- Single Source of Truth
- Behavior Driven Development
- Word < Wiki / Confluence < Markdown/TeX + Git
- Extract information (e.g. screenshots from acceptance tests, architecture from code, ...)
- Broken window theory (markdown linters, grammar check e.g. `write-good`, check changelog, ...)

# Revive your Architecture

de*PHP*end

- detects even the sneakiest dependencies
- generates dependency visualizations
- hackable (grep, sed, awk, ...)
- supports PHP 5.2 to 7.1

# de*PHP*end

## Features

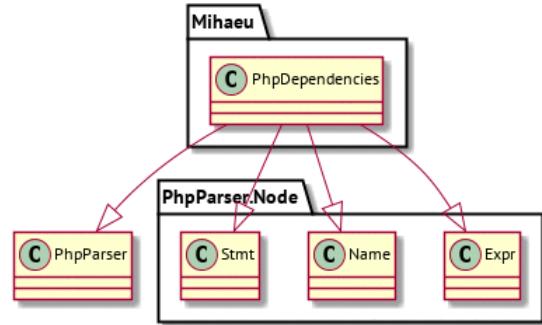
- **Text Output**  
For quick feedback, debugging, UNIX pipes etc.
- **Various Visualisations** (UML & DSM & dot)  
Detailed dependency & architectural analysis
- **(Metrics)**

# de*PHP*end

## How does it work?

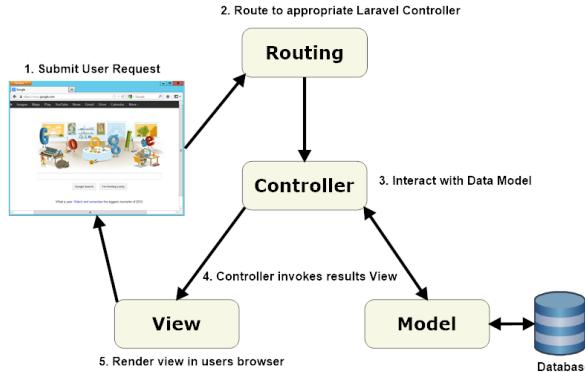
- parses PHP code
- builds an abstract representation (AST)
- detects dependencies for all classes
- applies filters and transformers
- writes output as text, pictures e.g.

# de*PHP*end





# dePHPend



```
<?php

$cmd = shell_exec('dephpend text src --no-classes');
$constraints = [
    'Model.* --> .*View',
    'View.* --> .*Model',
];
$regex = '/(.implode(') | (', $constraints).')/x';

if (preg_match($regex, $cmd)) {
    echo 'Architecture violation'.PHP_EOL;
    exit(1);
}
```

# ArchUnit

```
@ArchTest
public static final ArchRule layers_are_respected = layeredArchitecture()
    .layer("Root").definedBy("com.tngtech.archunit")
    .layer("Base").definedBy("com.tngtech.archunit.base..")
    .layer("Core").definedBy("com.tngtech.archunit.core..")
    .layer("Lang").definedBy("com.tngtech.archunit.lang..")
    .layer("Library").definedBy("com.tngtech.archunit.library..")
    .layer("JUnit").definedBy("com.tngtech.archunit.junit..")

    .whereLayer("JUnit").mayNotBeAccessedByAnyLayer()
    .whereLayer("Library").mayOnlyBeAccessedByLayers("JUnit")
    .whereLayer("Lang").mayOnlyBeAccessedByLayers("Library", "JUnit")
    .whereLayer("Core").mayOnlyBeAccessedByLayers("Lang", "Library", "JUnit")
    .whereLayer("Base").mayOnlyBeAccessedByLayers("Root", "Core", "Lang", "Library", "JUnit");
```



## Escaping Dependency Hell

# Summary

- Use your baloney detection kit
- be wary of agile "processes"
- let's all be architects
- fail fast and learn rapidly
- live architecture
- validate your architecture decisions
- revive your documentation

# Questions?

Michael Haeuslmann

@michaelhaeu

<https://github.com/mihaeu>

Feedback?

# Books

- **Clean Architecture** by Robert C. Martin
- **Agile! - The Good, the Hype and the Ugly** by Bertrand Meyer
- **The Pragmatic Programmer** by Andrew Hunt, David Thomas
- **Extreme Programming Explained** by Kent Beck, Cynthia Andres
- ...

# Sources

<http://www.denofgeek.us/sites/denofgeekus/files/6/25/frankenstein-1931.jpg>  
<https://www.slideshare.net/gahlawatanju21/agile-reluctancy-in-india-anju-gahlawat>  
<http://www.bradvg.com/blog/the-problem-with-good-intentions/>  
KurtKamka <https://www.getdpi.com/forum/leica/21559-s-show-us-your-s2-shots-22.html>  
<https://tisquirrel.me/2016/01/03/say-agile-one-more-time/>  
<http://www.br.de/themen/wissen/inhalt/umwelt/wueste-wuestentiere104.html>  
<https://blogs.microsoft.com/eugenio/2010/05/24/windows-azure-architecture-guide-part-1-release-candidate-documents-updated/>  
<http://eduart4kids.com/painting-for-kids/>  
<https://image.slidesharecdn.com/quangnguyencontinuousintegrationxpday2015danangquangnguyen-150520014341-lval-app6892/95/quang-nguyen-co>  
[https://blogs.microsoft.com/willy-peter\\_schaub/2010/11/22/are-the-rangers-courageous-or-crazy-to-brainstorm-and-adapt-methodologie](https://blogs.microsoft.com/willy-peter_schaub/2010/11/22/are-the-rangers-courageous-or-crazy-to-brainstorm-and-adapt-methodologie)  
<https://dotnetcodr.files.wordpress.com/2015/07/ddd-full-diagram1.png>  
[https://images-na.ssl-images-amazon.com/images/I/81vKL1%2BuEVL.\\_SL1500\\_.jpg](https://images-na.ssl-images-amazon.com/images/I/81vKL1%2BuEVL._SL1500_.jpg)  
<https://dotnetcodr.files.wordpress.com/2015/07/ddd-full-diagram1.png>  
[http://canarydwarf.co.uk/b/wp-content/uploads/2014/05/CD\\_questions\\_](http://canarydwarf.co.uk/b/wp-content/uploads/2014/05/CD_questions_)  
[https://www.batimes.com/images/wick\\_mar4\\_2.jpg](https://www.batimes.com/images/wick_mar4_2.jpg)  
<https://guidedpathedge.com/wp-content/uploads/2016/01/keep-calm-and-prioritize.png>