



# Agile Team Performance Measurement as a basis for accurate Cost Estimation

Harold van Heeringen  
Principal Consultant IDC Metri

# The Agile Challenge

01

Agile development is a challenge to manage and measure, especially compared to traditional development models like Waterfall

02

Management is often uncertain when functionality will be delivered and at what cost, its quality and risks inherent in the application

03

Key challenges experienced by business owners of agile products are:

- When will it be done
- How much will it cost
- Is the quality of the work produced sufficient
- How do I measure it and have actionable KPIs
- How can I communicate project success and value effectively to my business peers
- Is it stable
- Does it meet the organizational/customer requirements
- Will it be reliable and perform
- Can it scale
- Is it secure

04

- Due to the nature of agile (Story Points to assess progress), it is difficult to quantify agile
- If something is hard to quantify, it is hard to evaluate and manage.
- Still, Senior Management need to make decisions and demonstrate value delivered for budget spent.
- Software Cost Estimation is a necessity, also for agile projects!

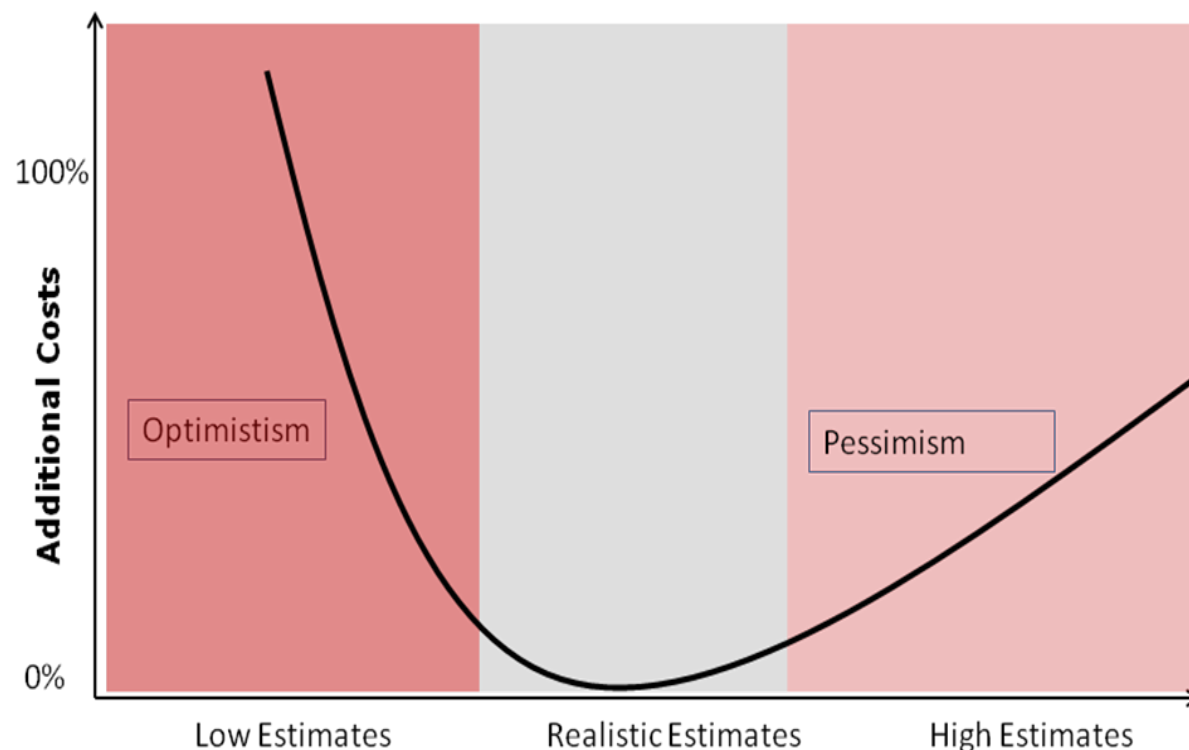
# Realistic estimation is a Key Success Factor

**A realistic estimate is one of the most important conditions for a successful project.**

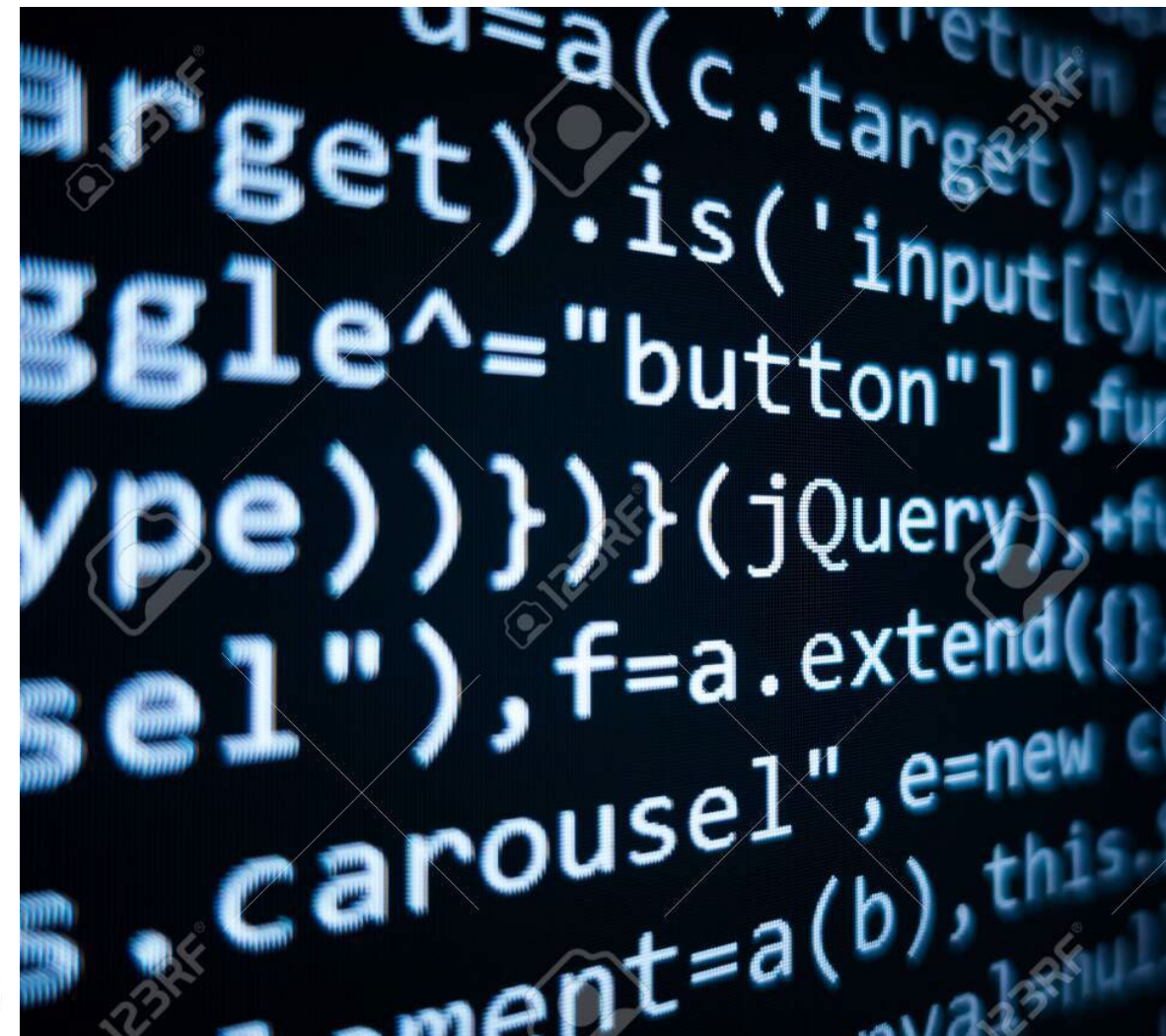
**The estimate is the basis for:**

- Business case
- Planning
- Proposal (outsourcing: fixed price / date)
- Financial result of the project... and the organization
- Claiming and releasing of resources
- Alignment between IT and business / customer
- Progress reports / dashboards and corrective actions
- The feeling of the team and the stakeholders
- Demonstrating Value for Money to stakeholders

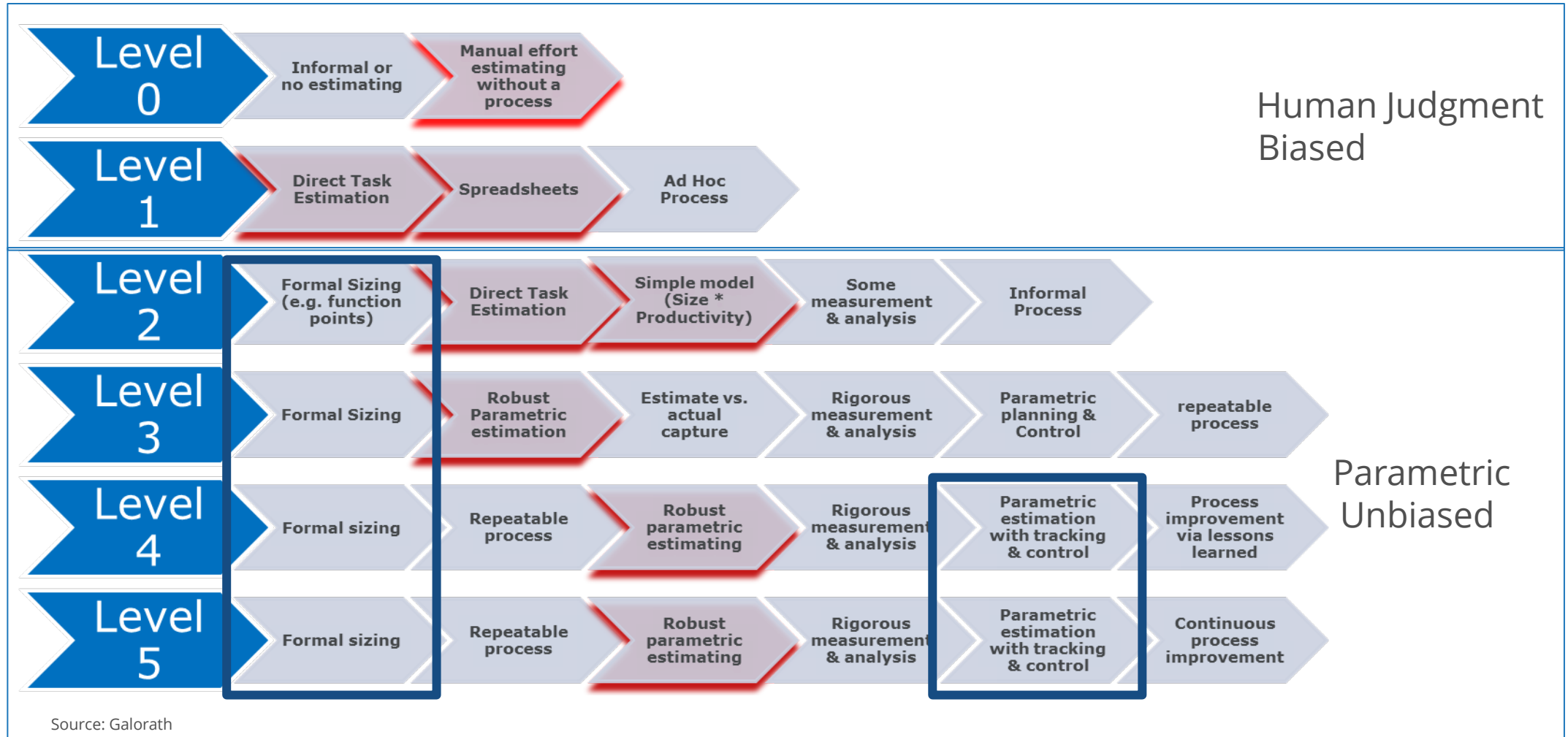
Without a realistic estimate, **the project is likely to fail!**



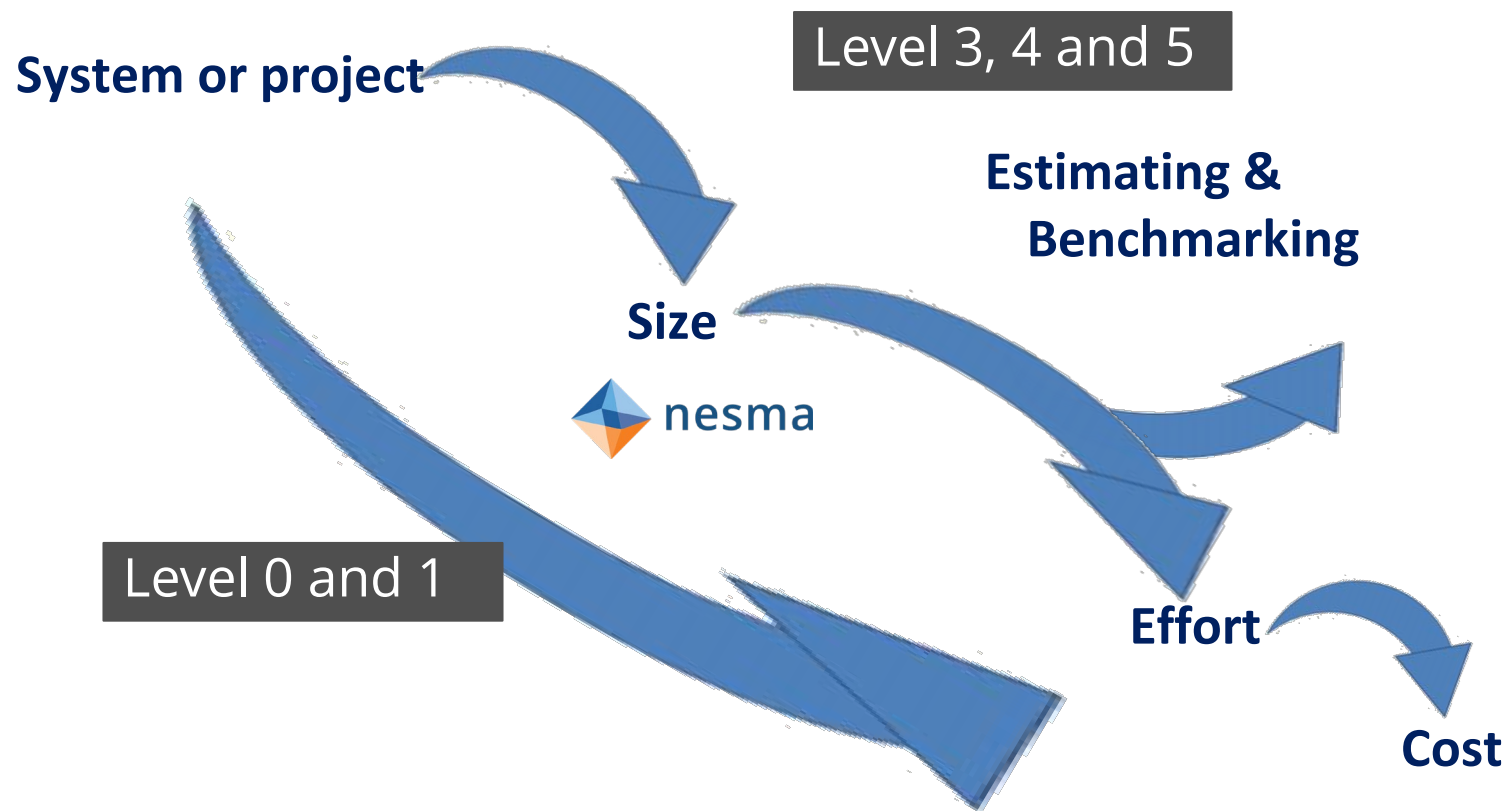
Software Cost Estimation is hard?!



# Software Cost Estimation maturity model



# Sizing allows estimation and benchmarking using data



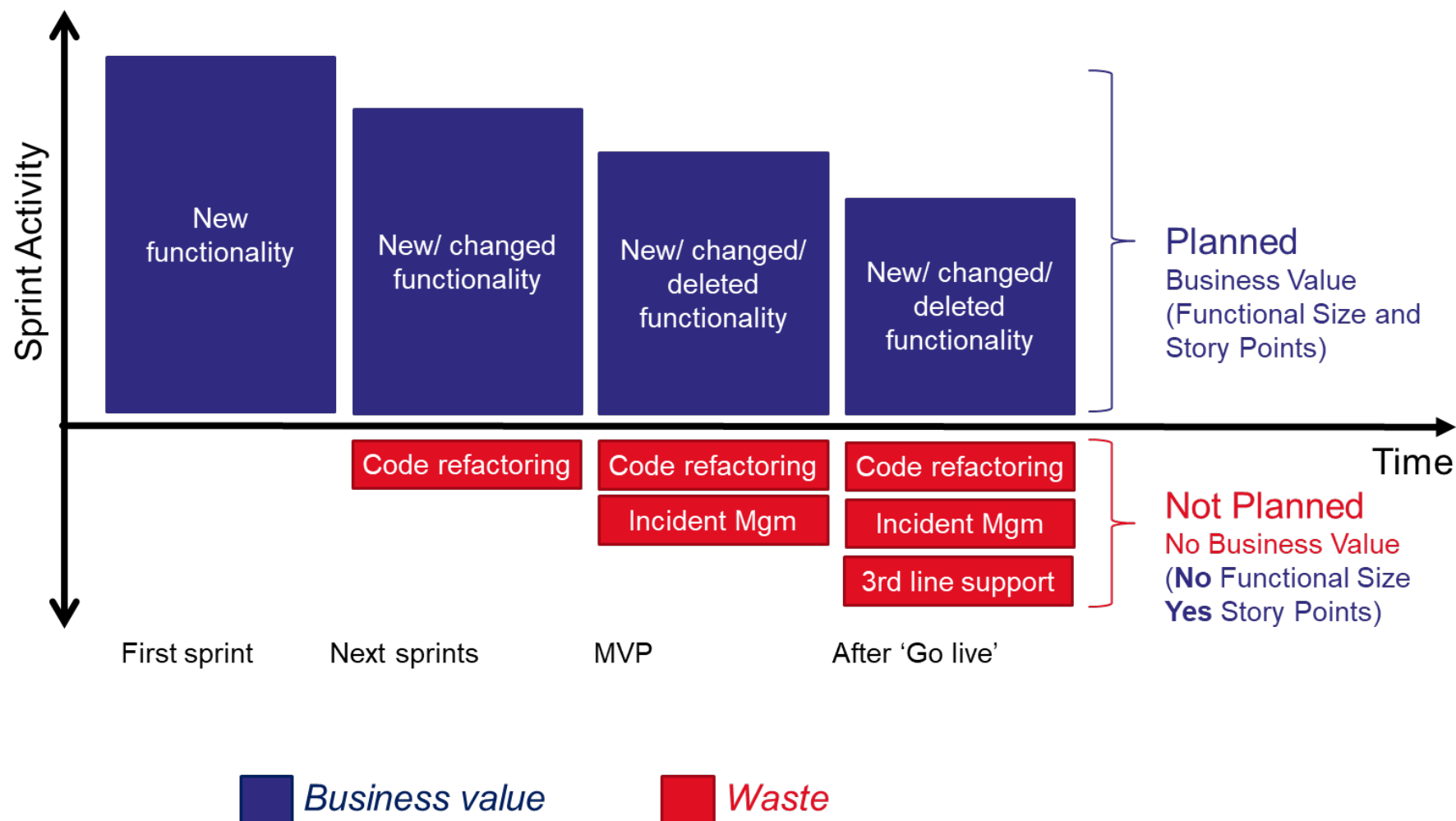
Metric	PDR (h/FP)
N	151
Min	2,1
P10	5,0
P20	6,1
P30	6,7
P40	7,2
Median	7,8
P60	8,7
P70	9,4
P80	11,2
P90	12,6
Max	76,6
Avg	10,6



1000 FP →

6700 – 7800 – 9400 effort hours

# Story Points versus Function Points



# Functional Size Measurement and Agile/Scrum

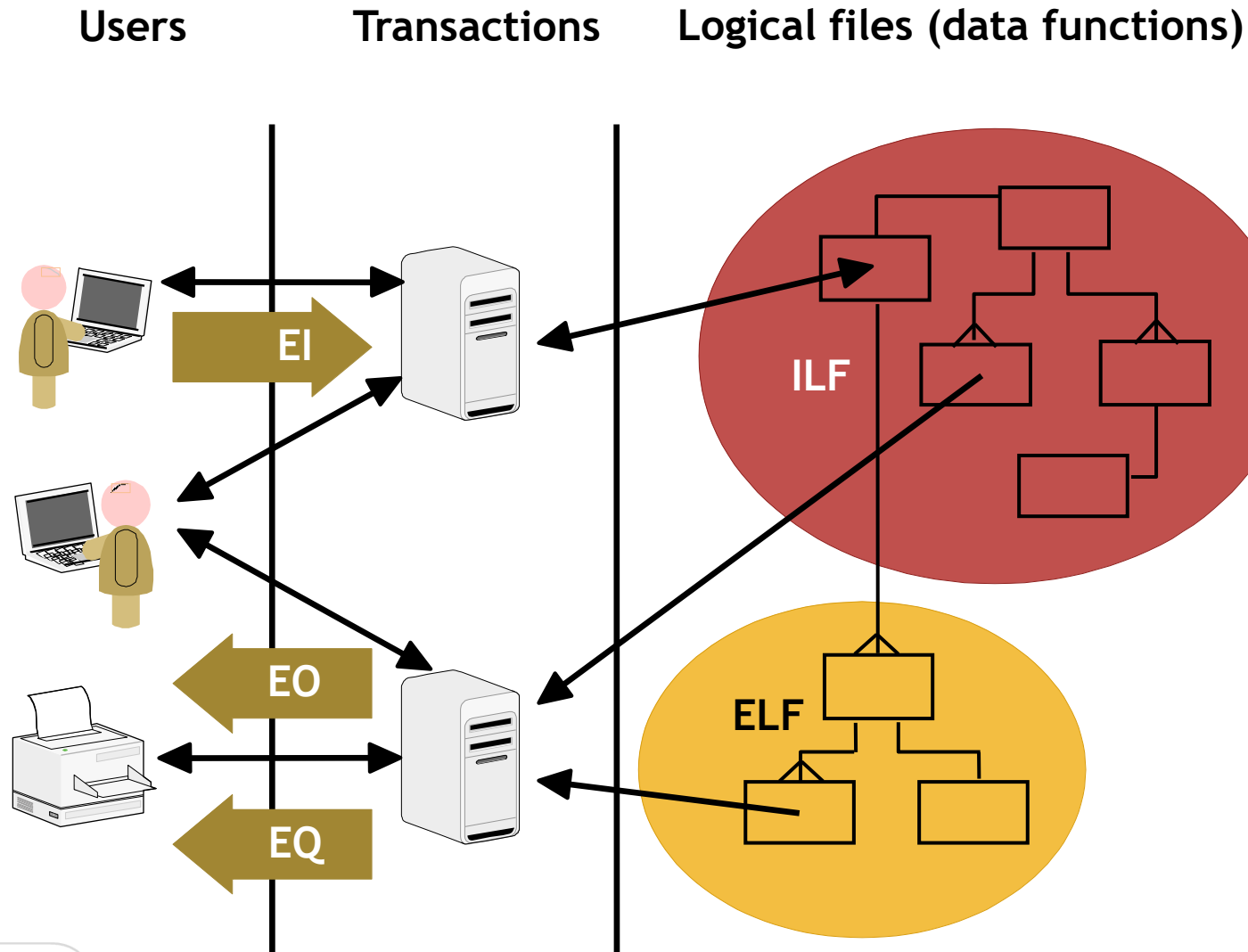


9/24





# Nesma ISO/IEC 24570: 2018

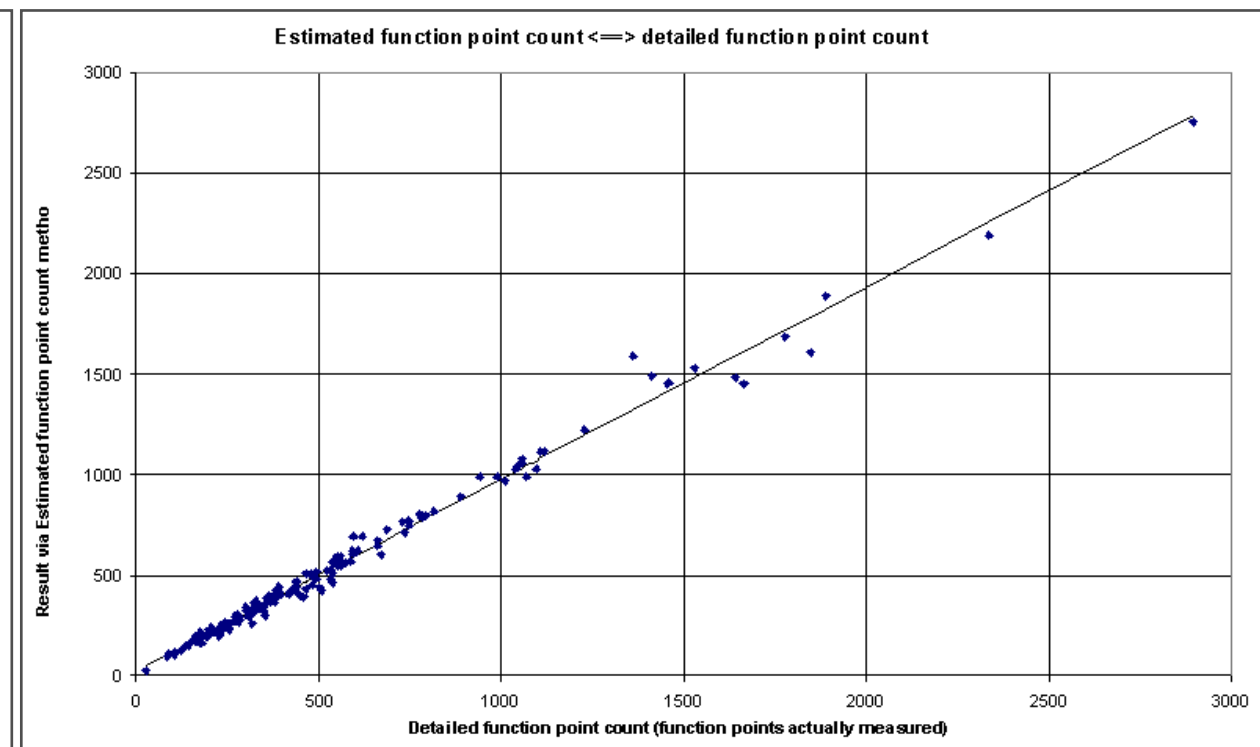
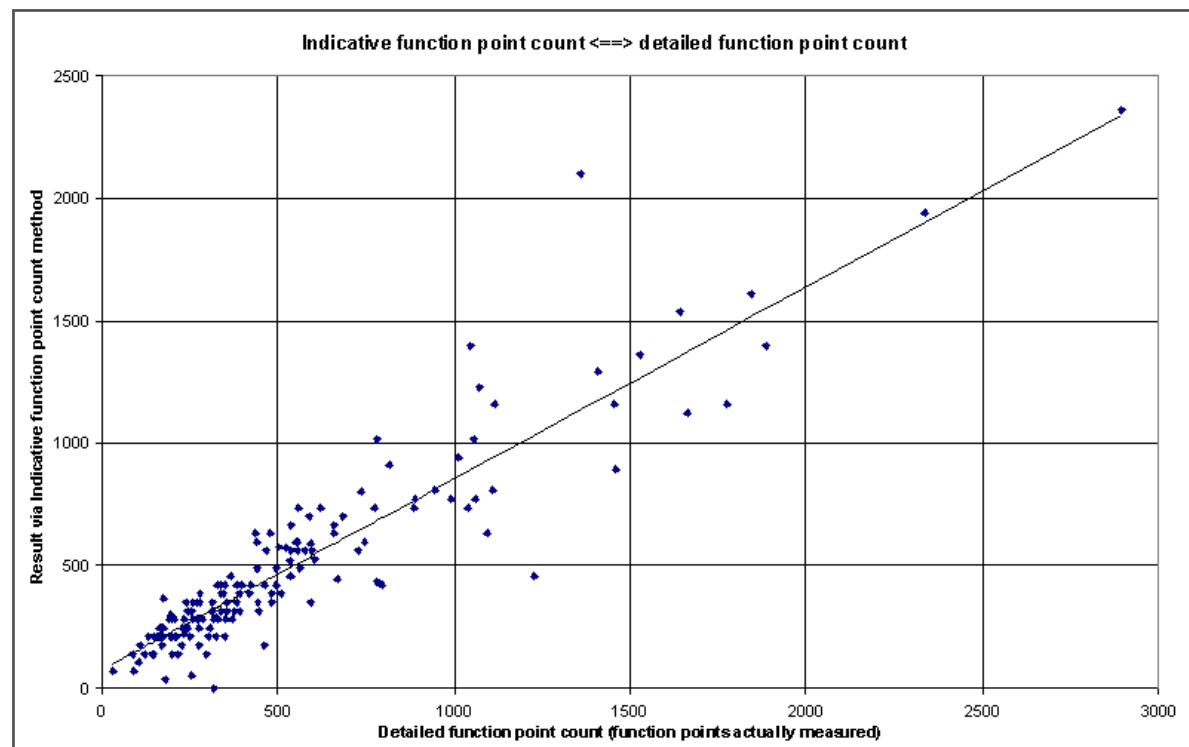


Detailed	Simple	Average	Complex
ILF	7 FP	10 FP	15 FP
ELF	5 FP	7 FP	10 FP
EI	3 FP	4 FP	6 FP
EO	4 FP	5 FP	7 FP
EQ	3 FP	4 FP	6 FP

Indicative	FP
ILF	35 FP
ELF	25 FP

Approximate	Simple	Average	Complex
ILF	7 FP		
ELF	5 FP		
EI		4 FP	
EO		5 FP	
EQ		4 FP	

# Nesma Indicative vs Approximate Functional Size Measurement



# User Stories are easy to measure

Example:

As a **customer** I want to **review my shopping cart** so **that I can decide if I need to add or remove items** before going to checkout.

Nesma Approximate FPA:

1 External Output = 5 Function Points

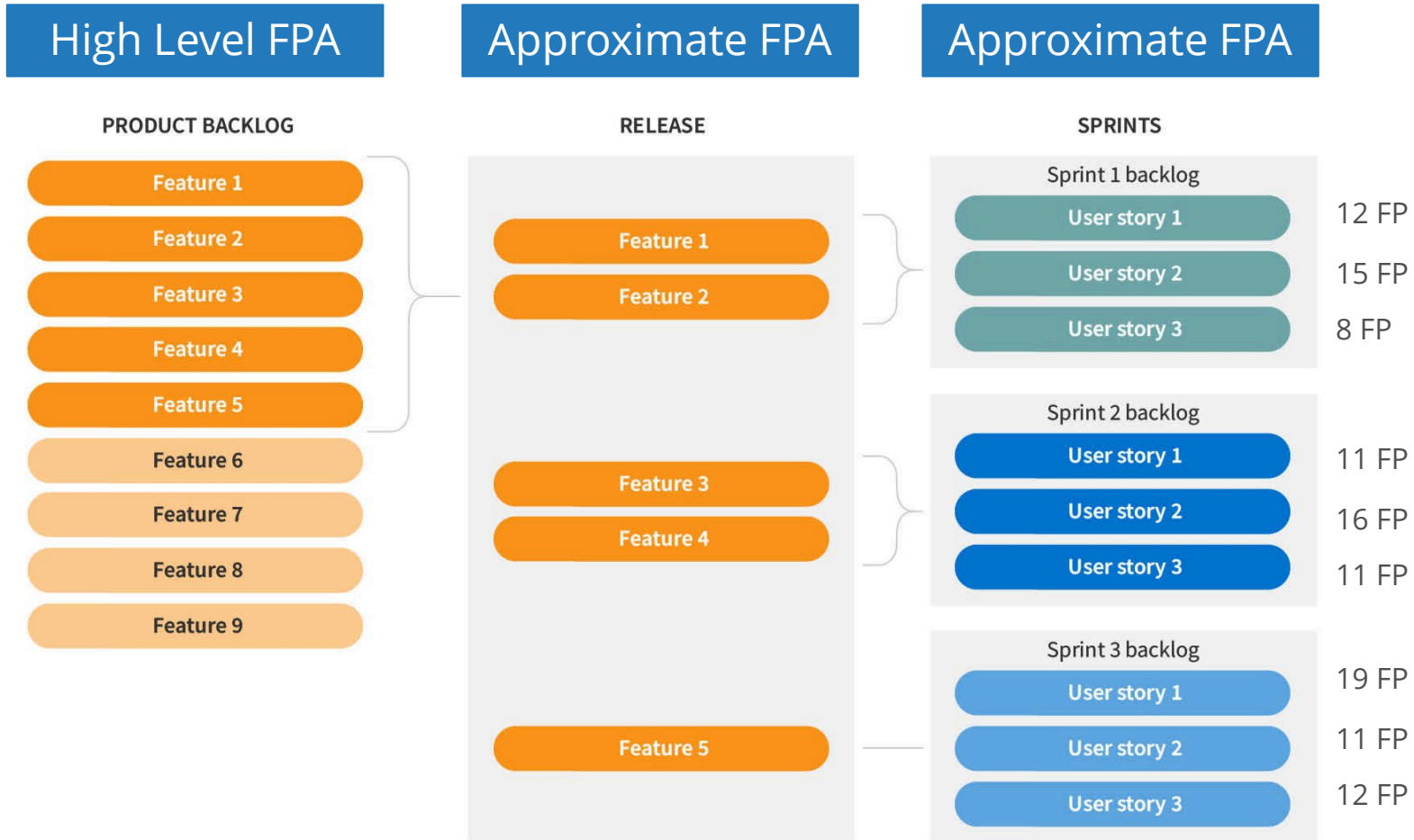
If not already created:

1 ILF Shopping Cart = 7 FP

## user story template

<b>WHO</b> are we building it for? Who is the user?	As a <type of user>
<b>WHAT</b> are we building? What is the intention?	I want <some goal or objective>
<b>WHY</b> are we building it? What is the value for the customer?	So that <benefit/value>

# Agile Project Estimation



5000 FP (+/- 30%)

500 FP (+/- 15%)

© 2020 Ahal Labs Inc.

# Simple High-level Example

Company starts to develop a new ERP system in .Net.

Indicative FPA measurement: 5000 FP +/- 30%

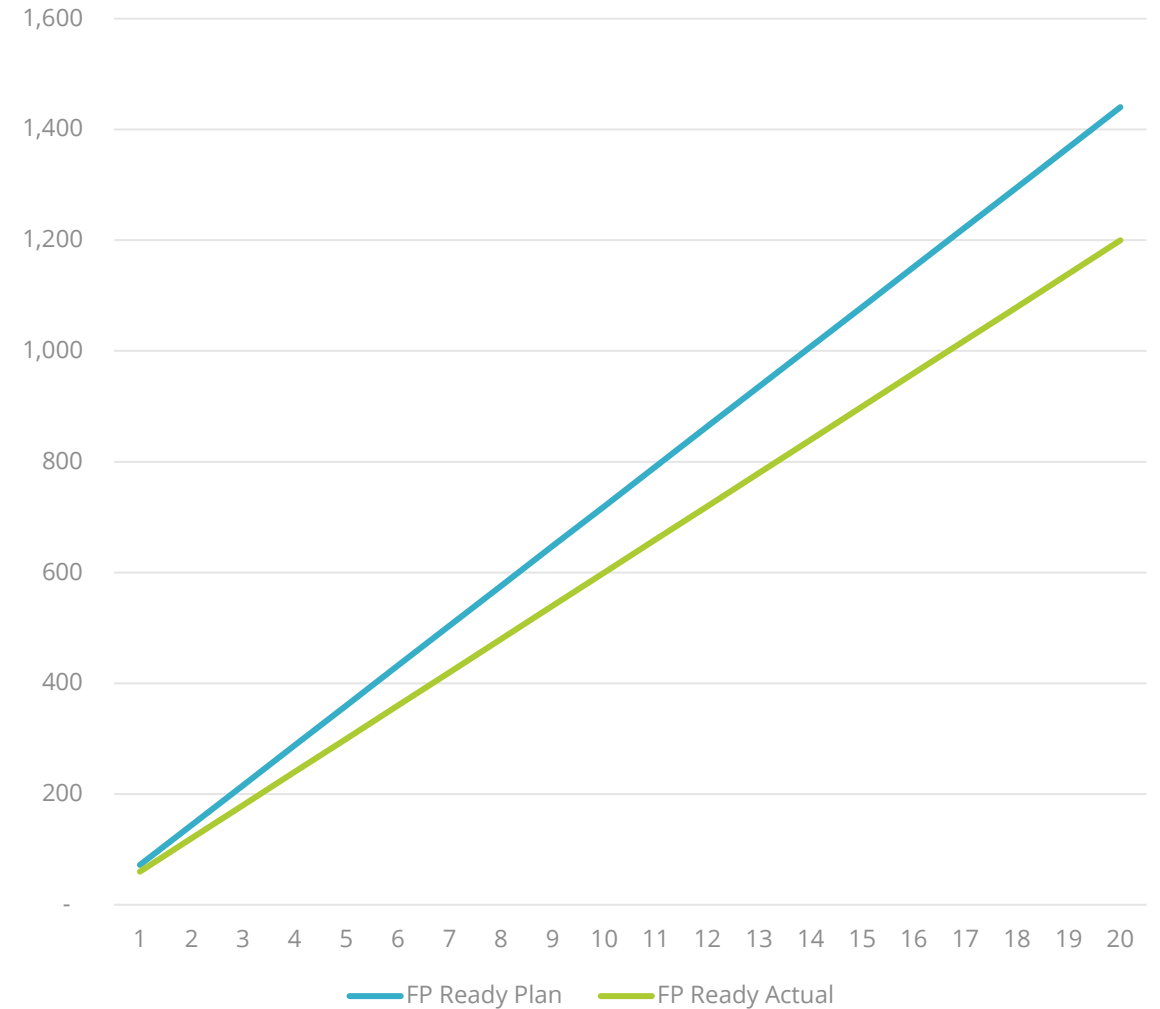
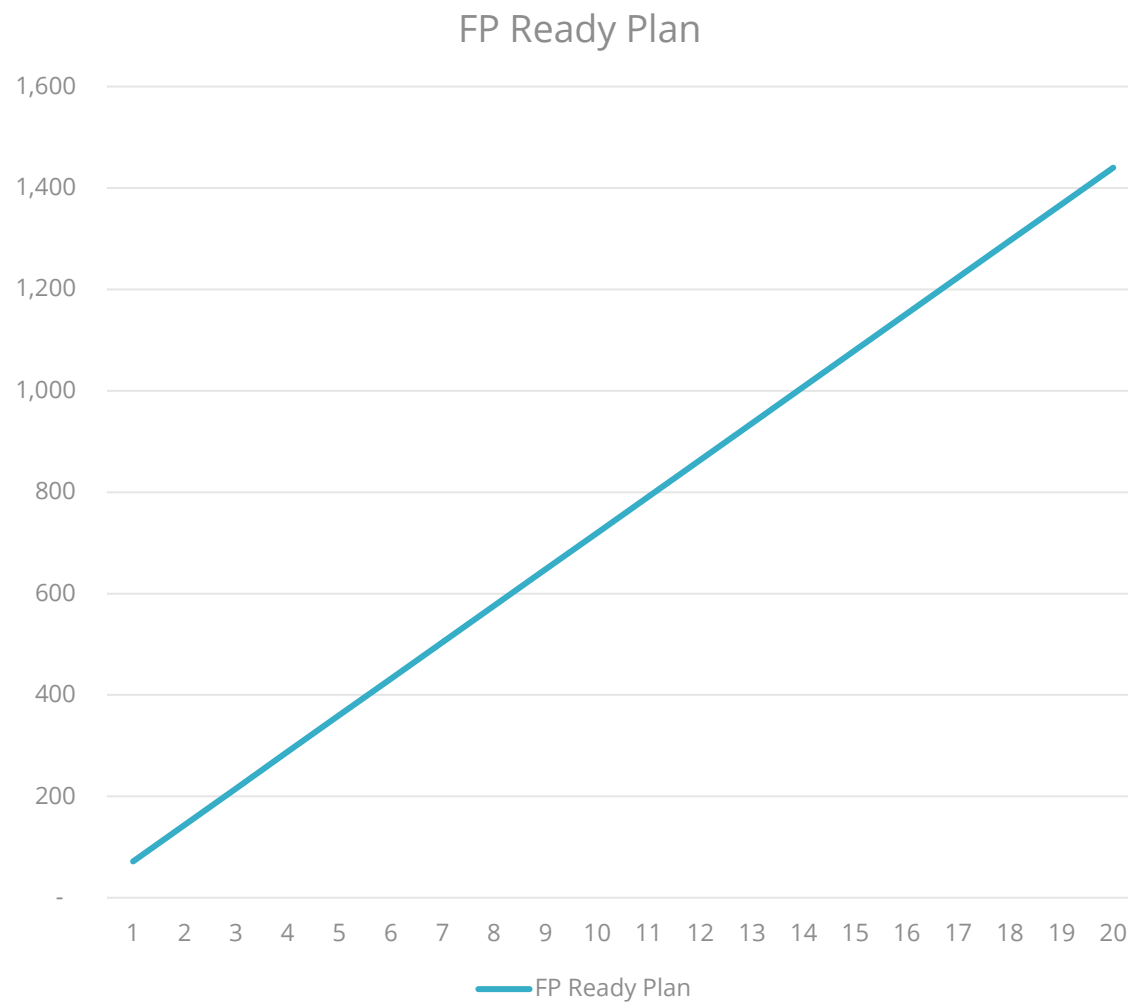
Metric	PDR (h/FP)
N	151
Min	2,1
P10	5,0
P20	6,1
P30	6,7
P40	7,2
Median	7,8
P60	8,7
P70	9,4
P80	11,2
P90	12,6
Max	76,6
Avg	10,6

Size	PDR	Hours	Cost	Sprints
5000	6,7	33.500	\$ 3.350.000	60
5000	7,8	39.000	\$ 3.900.000	70
5000	9,4	47.000	\$ 4.700.000	84

Team: 7 fte, 560 effort hours per sprint (2 weeks)

Blended rate: \$100

# High-level estimate → high-level planning



# Key metrics to measure and benchmark

Productivity = Effort hours spent per FP delivered

Cost Efficiency = Cost spent per FP delivered

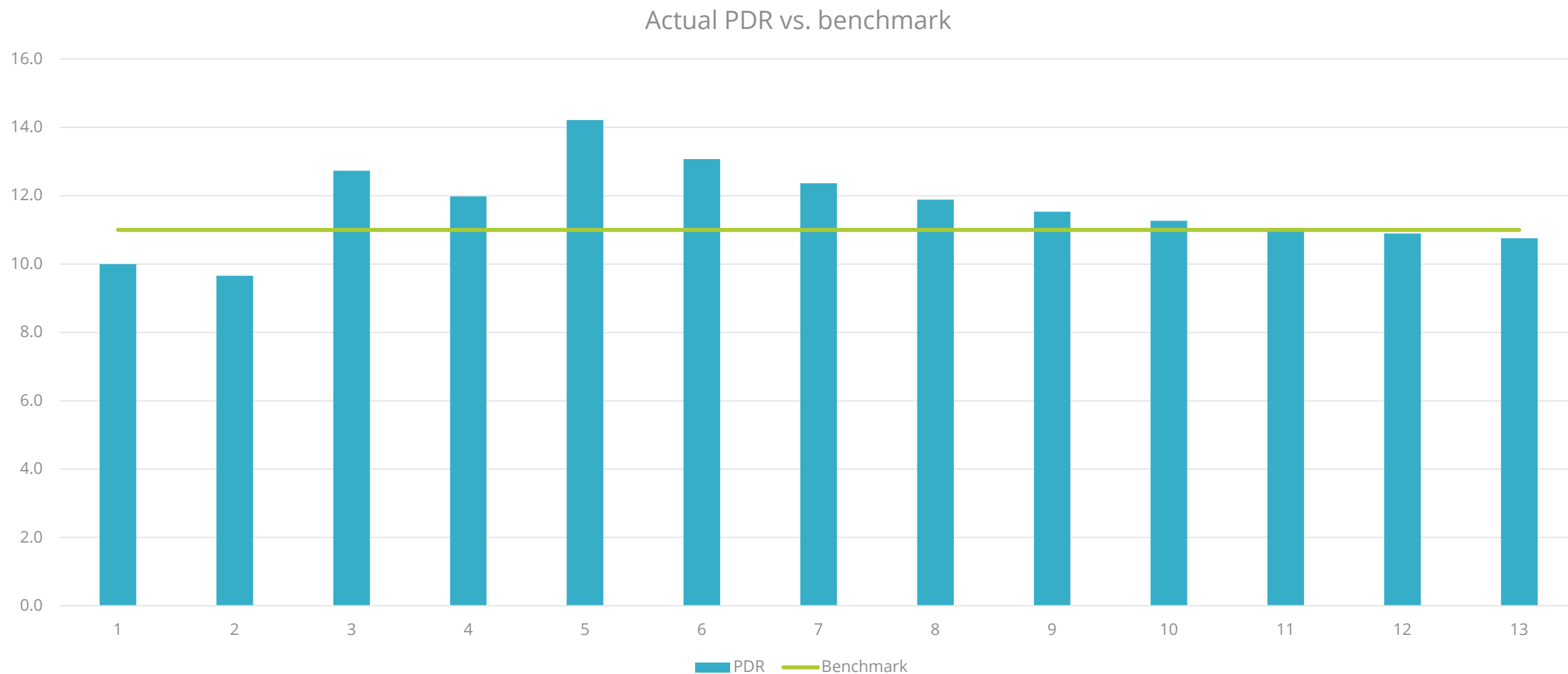
Delivery Speed = FP delivered per Month

Process Quality = Defects found per 1000 FP

Product Quality = ISO 25010 or ISO 5055 standards for software quality measurement.

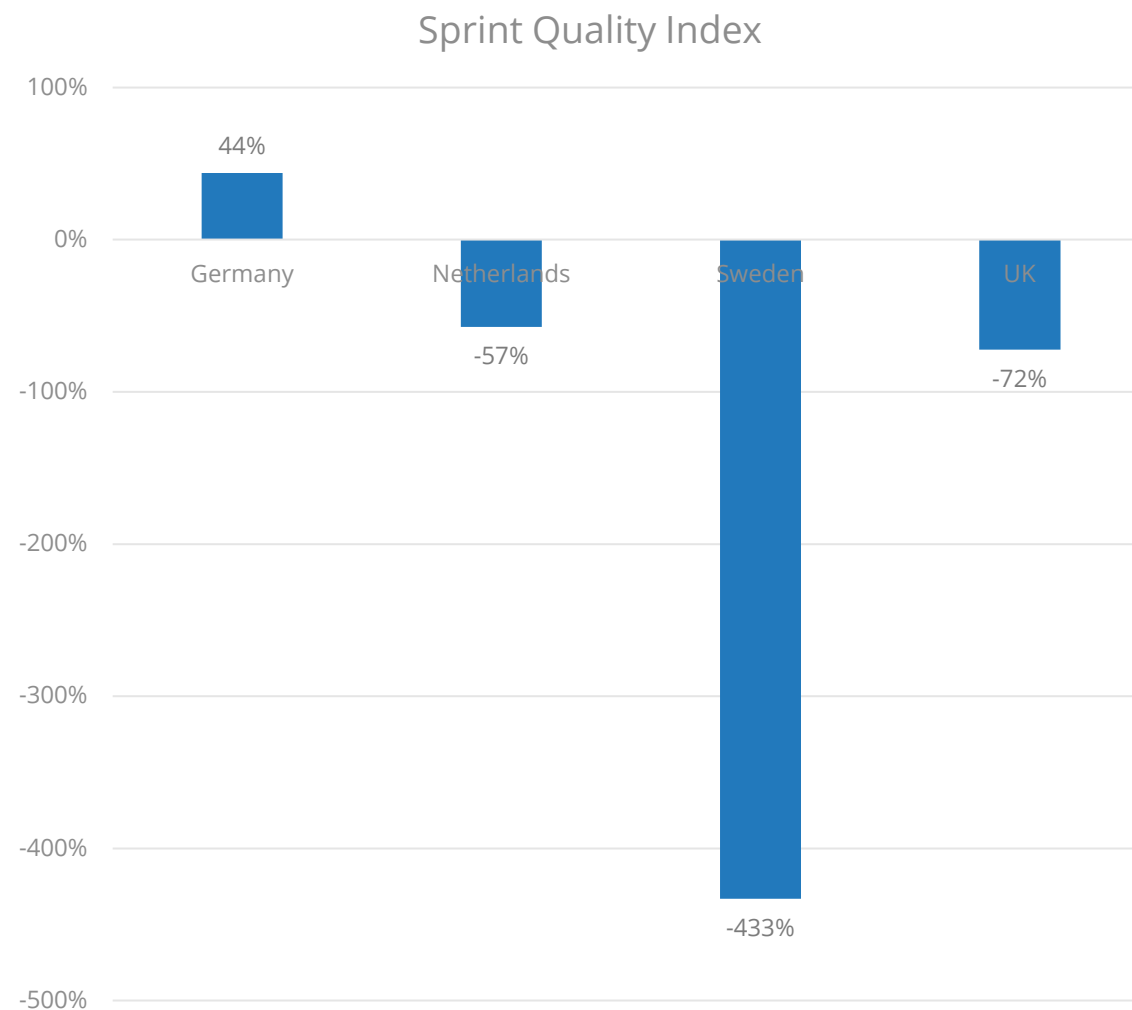
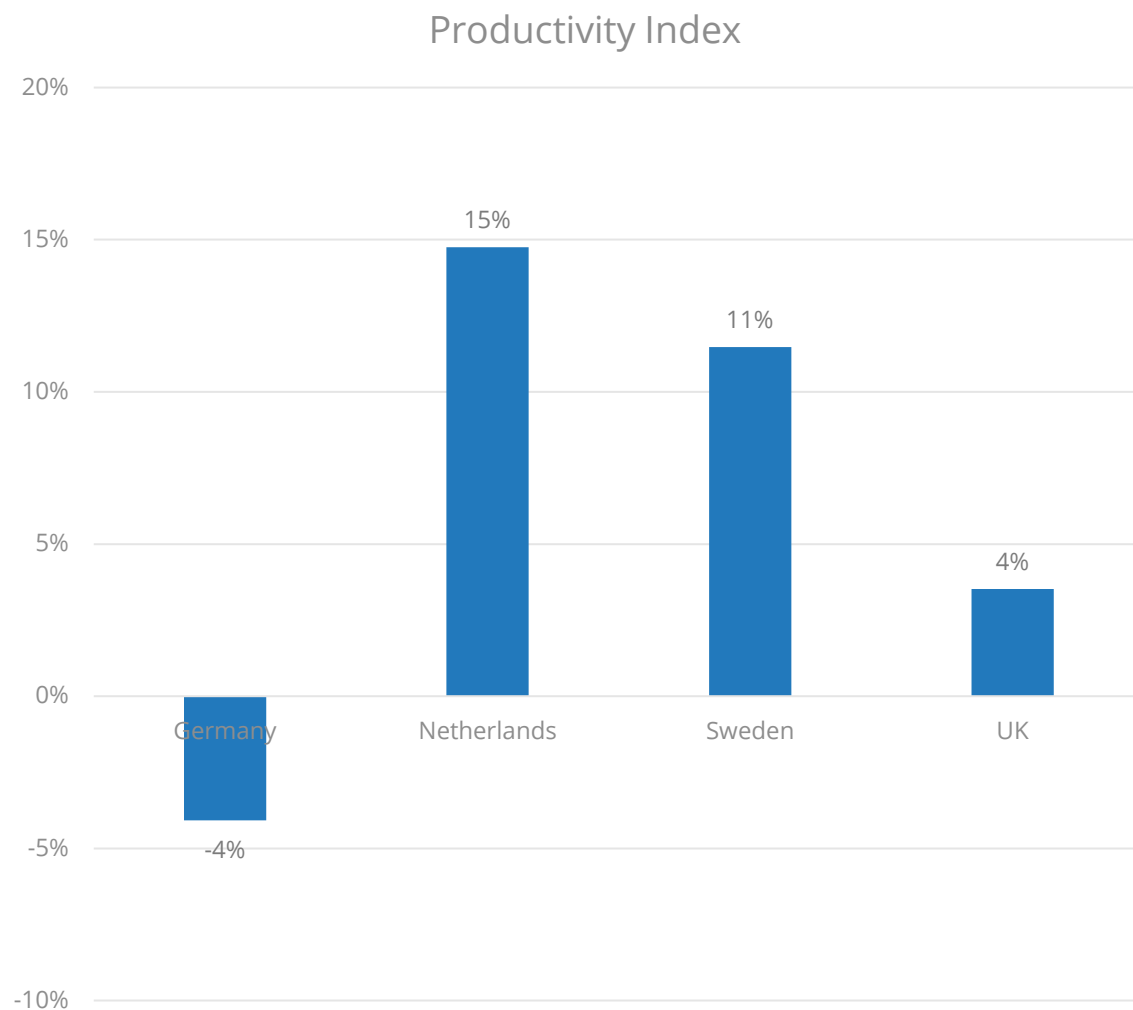
When you understand the performance of the teams, you can manage the project!

# Benchmarking

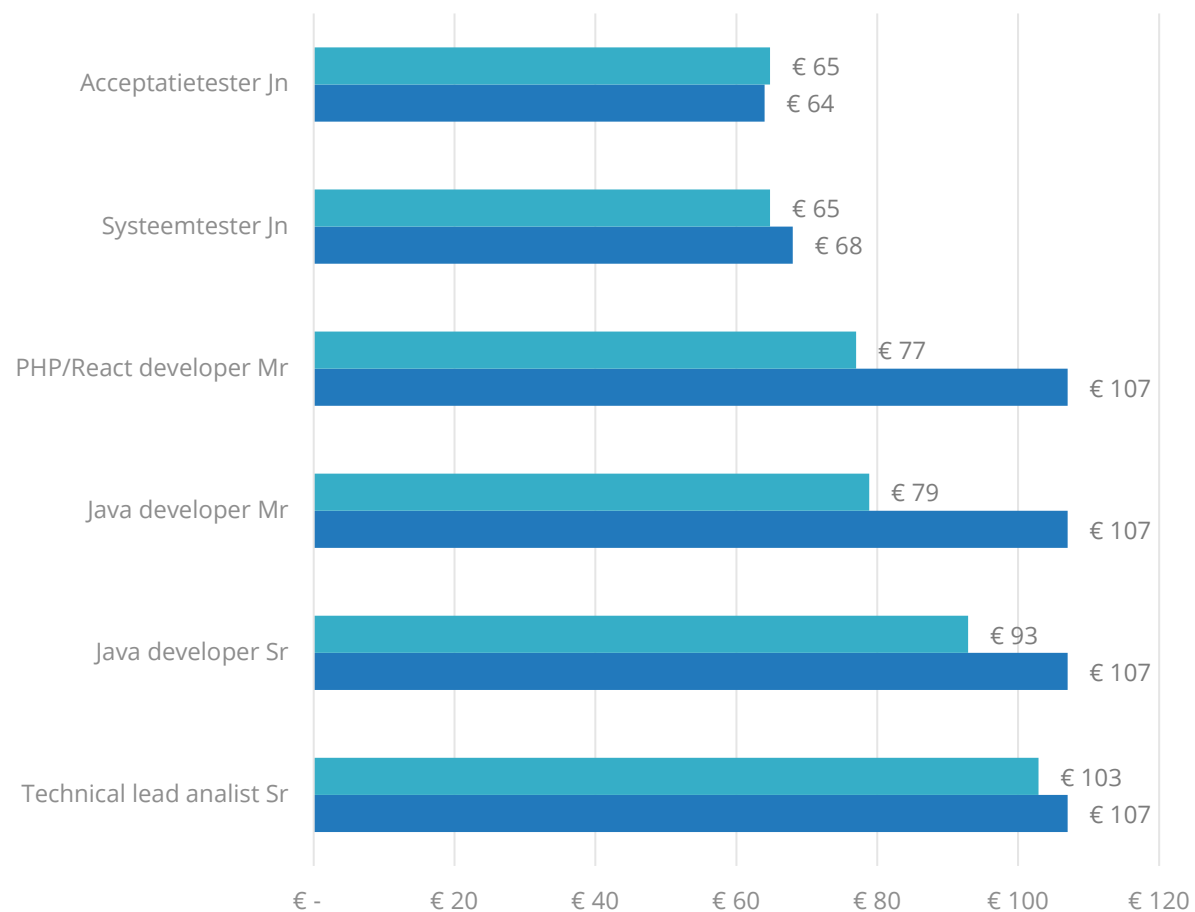




# 4 similar teams in 4 countries

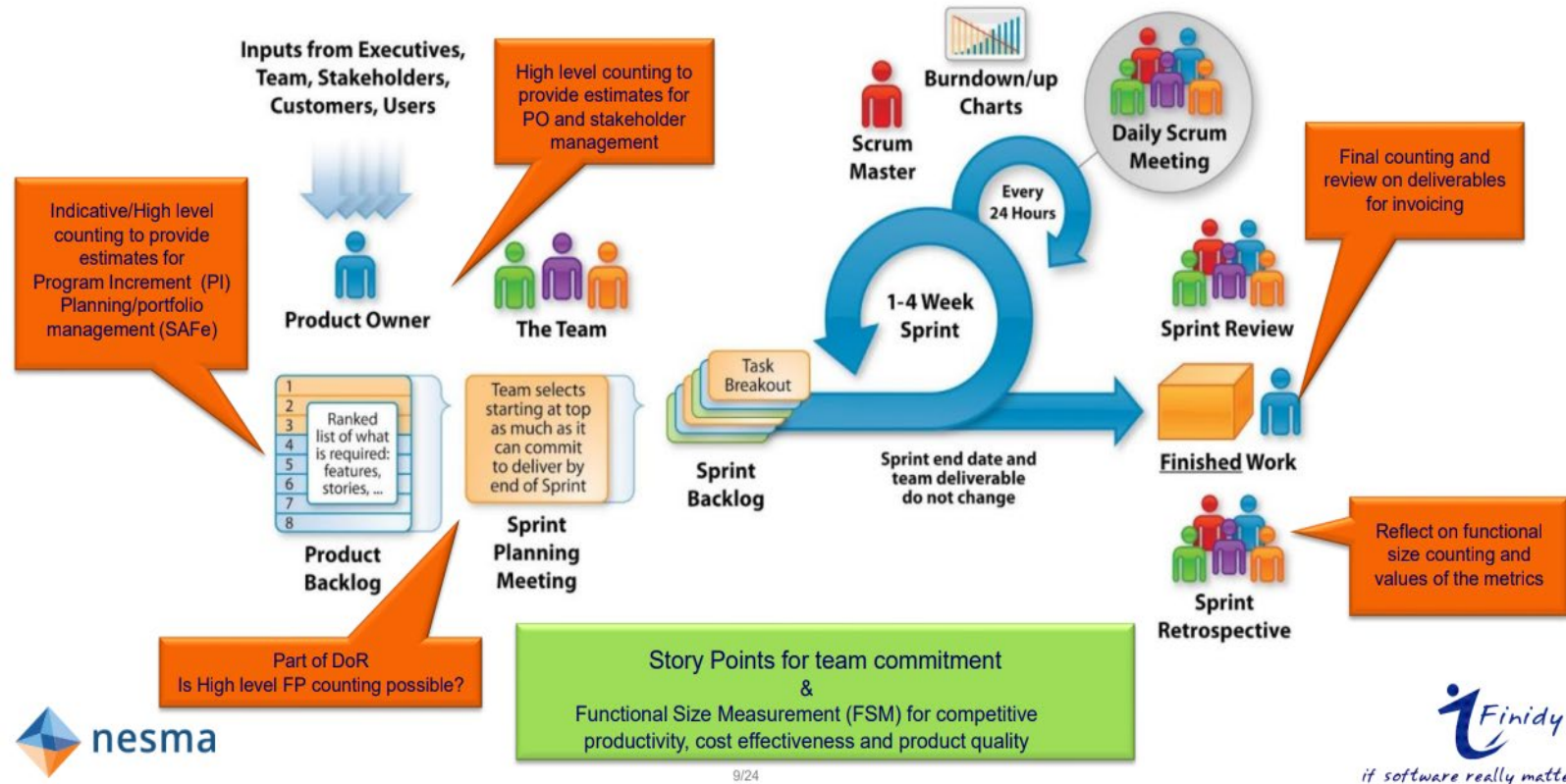


# From Effort to Cost – Hourly or Blended rates



Productivity drives cost, but hourly rates also.

# Functional Size Measurement and Agile/Scrum



Initial Estimation

Track Progress, Calculate metrics, Benchmark, and update Estimate

# Software Cost Estimation in an Agile world

## 10 key take aways

1. Software Cost Estimation seems hard, why is this?
2. Sizing is a crucial step
3. Software cost estimation is not a one-time activity.
4. There are objective Agile team metrics that can be compared to industry averages.
5. In all software cost estimations, there is an implied Productivity distribution.
6. Monitoring agile projects is important, as story point metrics are misleading.
7. Functional Size Measurement creates reference points.
8. Measuring progress based on functional size helps managing the project.
9. Especially in agile, beware of scope creep due to rework and/or requirements change. These will result in extra story points, but not in extra function points.





Harold van Heeringen  
Principal Consultant IDC Metri

[hvanheeringen@idc.com](mailto:hvanheeringen@idc.com)



[IDC.com](http://IDC.com)



[linkedin.com/company/idc](https://linkedin.com/company/idc)



[twitter.com/idc](https://twitter.com/idc)



[blogs.idc.com](https://blogs.idc.com)