



# THE CONCEPTS OF SIZE AND PRODUCTIVITY

IN AN AGILE WORLD

ICEAA conference, Phoenix  
June 2018

## INTRODUCING ME

- Drs. Harold van Heeringen, Senior Consultant ADM Benchmarking at **METRI**.
- International Software Benchmarking Standards Group (**ISBSG**) – President
- Netherlands Software Metrics Association (**NESMA**) – board member
- Common Software Measurement International Consortium (**COSMIC**) - Dutch representative in the International Advisory Council (**IAC**)
- Dutch Association for Cost Engineers (**DACE**) – working group parametric analysis
- **ICEAA** trainer of CEBoK chapter 12: Software Cost Estimation,
- **ICEAA** developer of SCEBoK module Benchmarking
- Speaker at many conferences on software measurement, estimation and benchmarking



[President@isbsg.org](mailto:President@isbsg.org)  
[harold.van.heeringen@metrigrp.com](mailto:harold.van.heeringen@metrigrp.com)



[www.linkedin.com/in/haroldvanheeringen](http://www.linkedin.com/in/haroldvanheeringen)



[@haroldveendam](https://twitter.com/haroldveendam)



[haroldveendam](https://github.com/haroldveendam)

ISBSG: [www.isbsg.org](http://www.isbsg.org)

Nesma: [www.nesma.org](http://www.nesma.org)

METRI: [www.metrigrp.com](http://www.metrigrp.com)

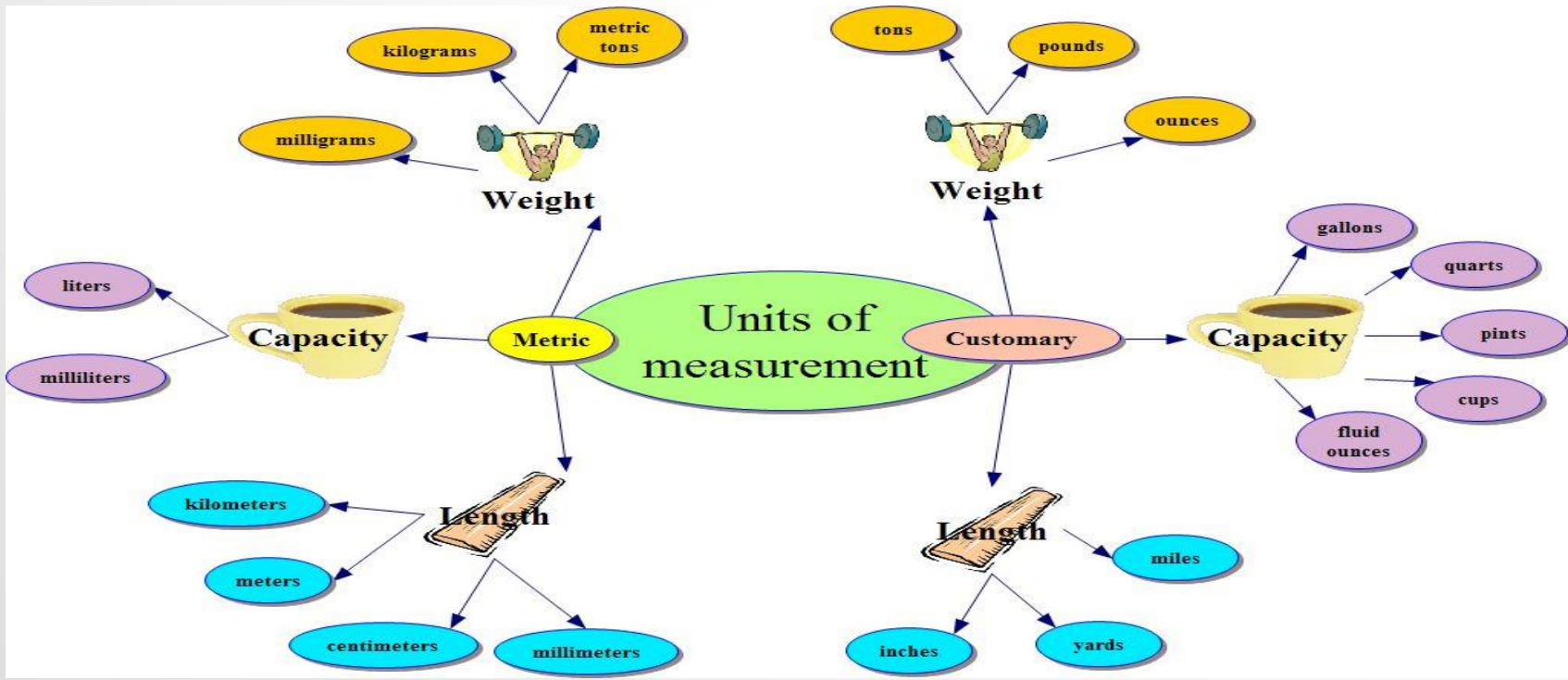
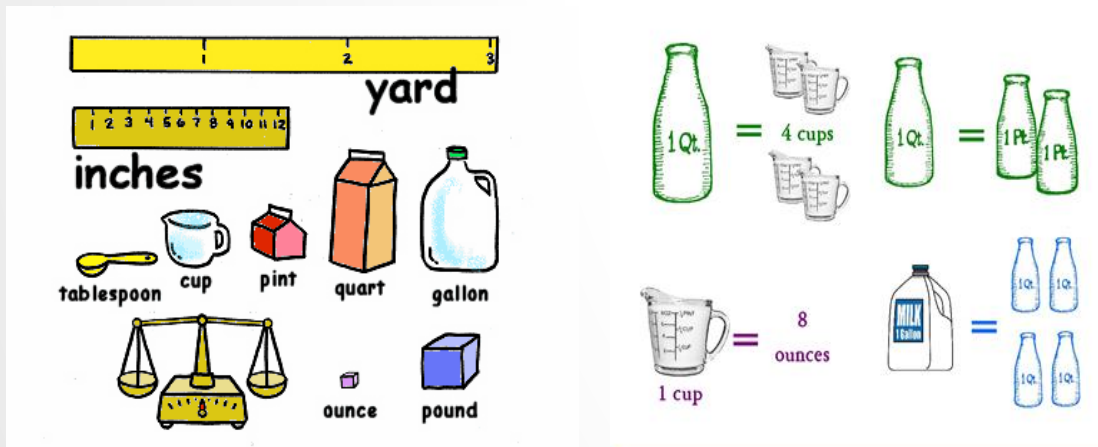


The global and independent source  
of data and analysis for the IT industry

## OVERVIEW

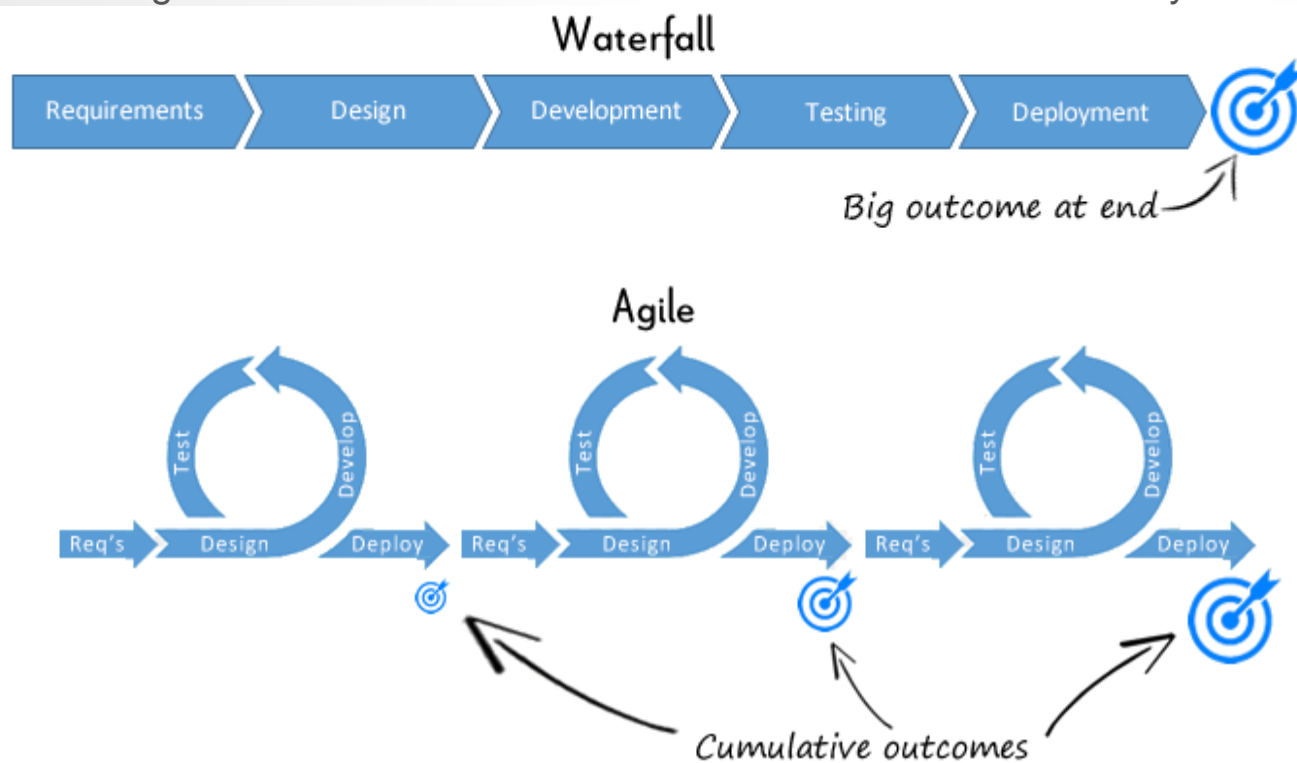
- Industry Maturity
- The concepts of Size and Productivity
- How are these concepts used in the IT world?
- The world is changing and becoming agile
- What is the effects of the agile world on Size and Productivity
- What are the implications of these effects?
- Recommendations for agile organizations and teams
- Example of agile team estimation
- Conclusions.

# WHY THIS PRESENTATION?



## CHANGES !

- The software development world is changing fast!
- From Waterfall development methods to Agile development methods.
- And Agile teams have different ideas about Size and Productivity!



## INDUSTRY MATURITY

- **Software industry: low maturity in performance measurement and estimation**
- Estimation and Performance Measurement processes are not targeted to software development and/or maintenance. Mostly financial metrics are used to measure performance.
- Organizations **don't know the size of their applications** and of their software portfolio.
- Organizations **don't know** if the **cost** spent on AD and AM is in line with industry averages.
- Organizations **don't know** their **productivity**.
- Organizations **don't know** their **time-to-market**.
- Organizations **don't know** their **cost efficiency**.
- Organizations **don't know** the **quality** of their software products.
- Result: Organizations don't know their **capability** compared to industry peers when it comes to **productivity, time-to-market, cost efficiency and quality**. They are not able to understand where they need to improve and not able to control process improvement.

## REASONS FOR LOW MATURITY

- Application Development is becoming more and more important for organizations as delivering new software functionality fast becomes more and more a driver for business.
- **Increasing performance is sometimes crucial for survival!**
- **Productivity** is the most important metric in most performance measurement processes as it is independent of locations (e.g. hourly rates). In general the notion that it is relevant, even crucial, to measure productivity is evident in almost all industries, except for the software industry. Why?
  - Productivity is universally defined by **output / input**.
    - Example: Painting a wall: **square meter** per **hour**.
  - **Input** is usually easily measured in AD and AMS: The number of effort hours spent per project or application.
  - But how do we measure **output**? What is the **size** of the software developed or maintained?

## OUTPUT MEASUREMENT

### Output is usually much harder to measure.

- The size of the software can be measured, however there are some issues with that. Software is **intangible** and can't really be measured with physical measures.
- Because it's not as evident to measure the output of software AD and AMS, many organizations don't have the knowledge, expertise and skills to do this. **Therefore the output is often not measured**, resulting in low maturity.
- **As output is not measured, there is no data about productivity. This results in low estimation maturity as well.**





## SOFTWARE SIZE

- How 'big' is the software to be developed, or maintained?
- Software is not physical, so how to measure it?
- Many attempts in the past:
  - Lines of Code - not standardized. Ambiguous.
  - Usecase Points - not standardized. Subjective.
  - Complexity Points - not standardized. Subjective.
  - IBRA points - not standardized. Subjective.
  - Nesda function points – **International standard: ISO/IEC 24570**
  - COSMIC function points – **International standard: ISO/IEC 19761**
  - IFPUG function points – **International standard: ISO/IEC 20926**

Most recently:

- Story Points - not standardized. Subjective. **Not a measure of size, but effort.**



## FUNCTION POINT ANALYSIS (FPA)

- Can be used early in the project, when functional requirements are known
- Independent of technical implementation. 500 FP Mobile app = 500 FP Legacy Cobol system
  - Just as a 20 m<sup>2</sup> glass wall = 20 m<sup>2</sup> brick wall
  - Effort to realize the software depends on **productivity**
  - Cost depends on **productivity** and **labor rates**.
- Independent of the systems requirements
- **Objective, verifiable, repeatable, defensible measurement !!**
- More function points means more functionality: **(business) value!**
- Functional size is the basis for **objective software metrics**:
  - Productivity (Hours per FP)
  - Cost Efficiency (Money per FP)
  - Time to Market (FP per calendar month)
  - Quality (Defects per 1000 FP)

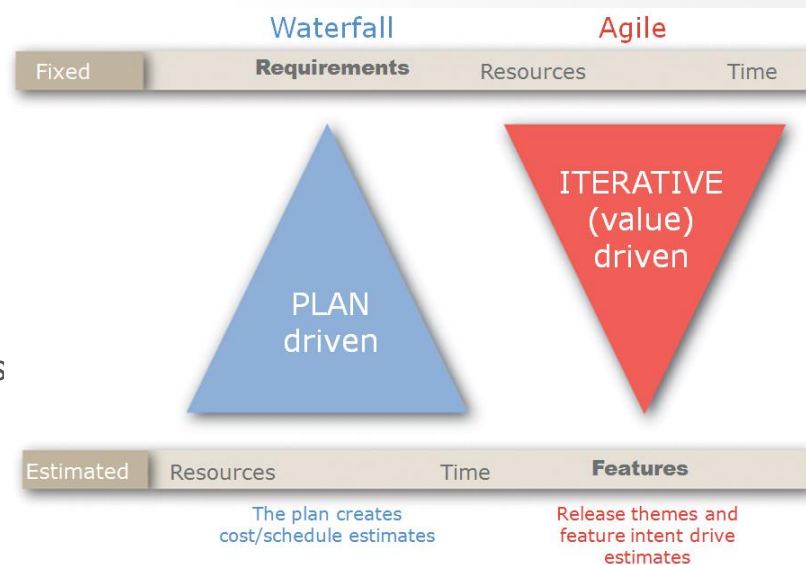


## HISTORY OF FUNCTIONAL SIZE

- 1970's: FPA developed by IBM to prove higher productivity of new programming languages and compilers. FPA published in the public domain in 1978.
- 1980's: Increased use of FPA, start-up of user groups (IFPUG (1986), NEFPUG (1989), etc.)
- 1990's: FPA adopted mainly in large organizations and large software development companies. In 1997, the ISBSG was founded.
- 2000's: FPA used in productivity measurement, estimation, benchmarking, price/FP contracts, supplier performance measurement.
- 2010's: **Decreasing use** of Functional Size Measurement.
  
- Question: Why is Functional Size Measurement use decreasing?
- One Answer: The world is becoming **more and more agile** !

## DECREASING USE OF FPA – WHY?

- The world is becoming **Agile!**
- 2001: **The Agile manifesto**: based on their combined experience of developing software themselves and helping others do that, the seventeen signatories to the manifesto stated that they value:
  - **Individuals and Interactions** more than processes and tools
  - **Working Software** more than comprehensive documentation
  - **Customer Collaboration** more than contract negotiation
  - **Responding to Change** more than following a plan



- Many people in the Agile community considers FPA to be a ‘*thing of the past*’ which can’t be used to measure the lightweight ‘user stories’ documentation.
- Therefore a new unit of measurement was created: **Story Points**.
- **But... Is it really not possible to measure user stories with FPA?**

## EXAMPLE

Typical example of a simple User Story:

- As a **book buyer** I want to **search for books by author name** to **see if any new books of the author have been published**.
- This user story can easily be measured in High-Level Nesma FPA:
  - 1 External Output function. 5 FP
  - 1 ILF Book: 7 FP
  - **Total: 12 FP**

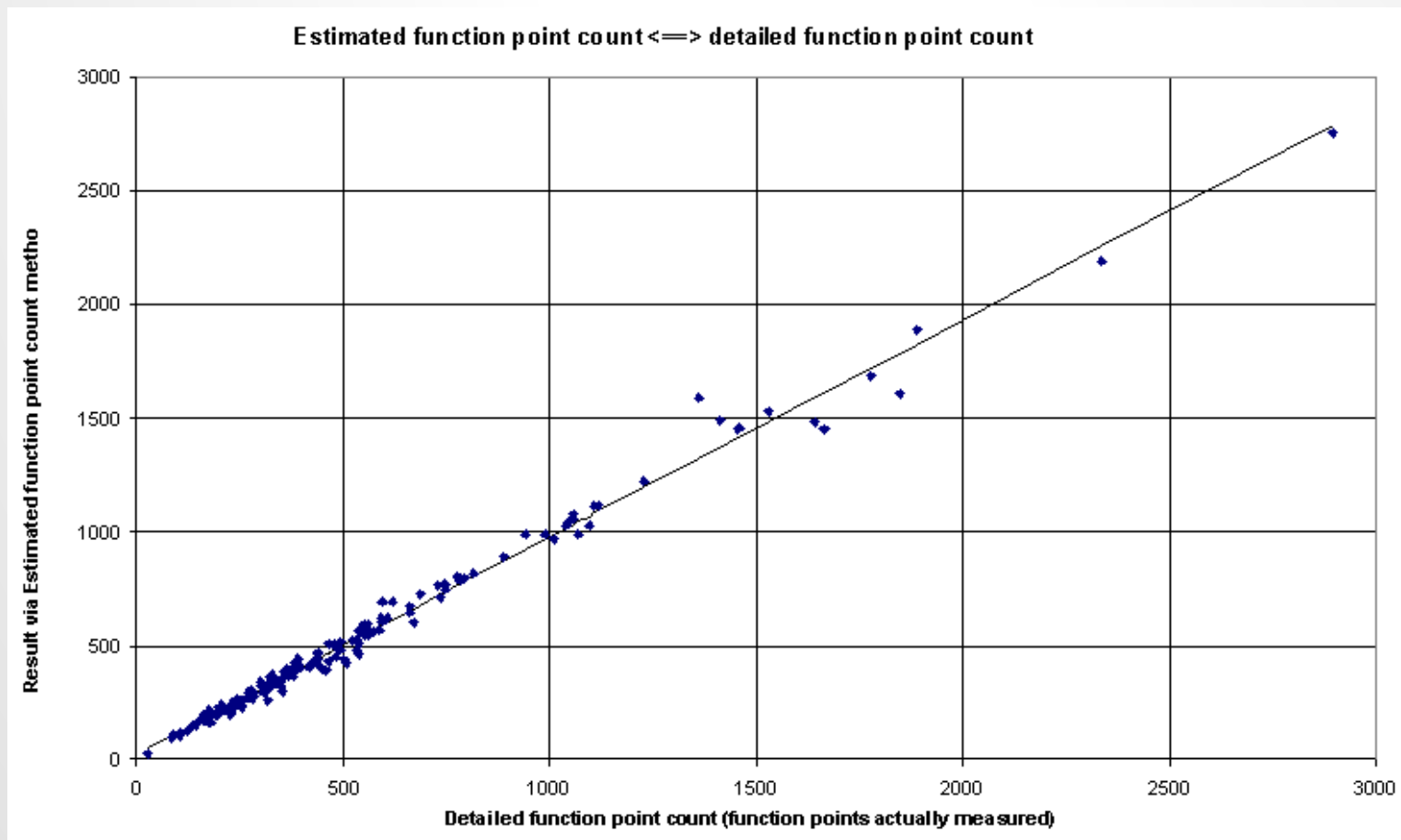
The story points assigned by the team may be 0, ½, 1, 2, 3, 5, 8, 13, et cetera, depending on the work they think needs to be done to implement this story.

For the user however, 12 function points of functionality are added to the application.

It's really that easy!



## ACCURACY OF NESMA HIGH-LEVEL FPA



<https://nesma.org/themes/sizing/function-point-analysis/early-function-point-counting/>

## STORY POINTS

- Story point is a **arbitrary measure** used by Scrum teams. This is used to measure the **effort** (not size) required to implement a story.
- In simple terms its a number that tells the team how hard the story is. Hard could be related to complexity, Unknowns and effort.
- Not comparable outside the team!
- Analogue example to assess a number of dog points to the size of a dog.

Dog	Dog points
Poodle	3
Schnauzer	3
German shepard	5
Chihuahua	1
Labrador	6
St Bernhard	6
Bulldog	4

Team X: German Shepard = 10  
 Team Y: Schnauzer = 10  
 Team Z: Chihuahua = 1

Dog points/Story points is **not a standard**  
 Not useful for management decision making  
 Not useful for collecting historical data  
 Not useful for project estimation  
 Not useful for benchmarking

Useful for sprint planning  
 Useful for velocity/burn down

## STORY POINTS ARE USEFUL ON THE TEAM LEVEL

- Story points is not a standardized metric for software size
  - Arbitrary metric
  - Effort metric not size metric
- **Therefore SP does not measure the output of software development teams**
- **On a team level, SP metrics are useful**
  - Before the sprint, Story points help to estimate the number of product backlog items that can be delivered based on experience of velocity (story points per sprint)
  - After the sprint, the actual story points delivered can be used to calibrate the velocity metric over the sprints, which then can be used to estimate more accurately which product backlog items can be delivered in the next sprints.
- **For management however, Story points are a step back in maturity.** Story point metrics don't give management information regarding:
  - Team size necessary (e.g. is the team large enough to deliver all necessary functionality on date dd-mm-yyyy)
  - Progress against targets (when will the minimum viable product go live)
  - Risk mitigation in outsourcing. How to manage supplier performance against contractual agreements.



## CURRENT SITUATION

- More and more organizations are moving towards agile and DevOps development teams
- These organizations usually stop functional size measurement and start story point metrics in their agile journey, often advocated by agile coaches and specialists.
- At some point, management loses grip and lacks the necessary metrics for decision making. They need to believe the teams on their word.
- Especially when application development is outsourced to supplier organizations, grip becomes less. Agile teams are usually contracted on a time and material basis, which is very beneficial for the supplier organization, as the risk is shifted back to the customer organization.
- Organizations like Nesma, IFPUG, COSMIC and ISBSG are trying to convince the industry that functional size measurement is still necessary, even in agile teams! <sup>1</sup>
- The industry needs to become more mature! Use standards, not arbitrary units!

<sup>1</sup> <https://nesma.org/2017/07/taming-agile-bandwagon/>

## STRONG RECOMMENDATIONS

- **Agile teams need to use story points on a team level**
  - Teams are used to these
  - Help Product Owner prioritize and plan the coming sprints
  - Commitment of the team members
  - Splitting user stories when too big
- **Functional size needs to be measured as well!**
  - Objective measurement of the functional size delivered to the user
    - **Business Value!**
  - Necessary for management processes
    - Long-term estimation
    - Track forecast against plan
    - Productivity Measurement
    - Supplier Performance Measurement
    - Benchmarking
- **How?**
  - Use Nesma High-Level FPA or COSMIC High-level method
  - Easy to learn
  - Easy to apply to User Story documentation

## EXAMPLE: LONG TERM ESTIMATION<sup>1</sup>

- The example: An organization needs to deliver a software product in 6 months and a list of high-level user stories that need to be realized ('Must haves') is provided.
- Using the high-level Nesma FPA method, the Scrum master is able to determine the functional size of the functional user requirements: 1300 Nesma function points. The reported accuracy of this method -8% to +15%.
- So the functional size of the product that needs to be realized by the team is:
  - Min: 1200 FP
  - Likely: 1300 FP
  - Max: 1500 FP
- Then we need relevant historical data for the next step, for example ISBSG data.

<sup>1</sup> Download the tutorial from the ISBSG agile corner: <http://isbsg.org/isbsg-agile/>

## INTERNATIONAL SOFTWARE BENCHMARKING STANDARDS GROUP (ISBSG)

- **Independent and not-for-profit;**
- Full Members are non-profit organizations, like China SPI, NESMA, IFPUG, Beijing Kexin Science and Technology Ltd, JFPUG, GUFPI-ISMA, FiSMA, and Swiss-ICT.
- Grows and exploits two open repositories of software data (.xls):
  - New development projects and enhancements (> **8000 projects**);
  - Maintenance and support (> **1200 applications**).
- **Everybody can submit project data**
  - Questionnaire on the site / on request (.xls) / online
  - Anonymous
  - Free benchmark report in return

# ISBSG DATA

>8000 rows in Excel, Easy to analyze.

>250 data fields (columns) per project

ISBSG Delivering IT Confidence													
D&E Release April 2016		8261 rows											
ISBSG Project ID	Rating	Rating	Software Age	Major Grouping	Major Grouping	Major Grouping	Major Grouping	Major Grouping	Major Grouping	Major Grouping	Major Grouping	Sizing	Sizing
	Data Quality Rating	UFP rating	Year of Project	Industry Sector	Organisation Type	Application Group	Application Type	Development Type	Primary Programming Language	Count Approach	Functional Size	Relative Size	
10019	B	B	2014	Communication	Telecommunicati	Business Applicati	Data Warehouse	Enhancement	Shell	IFPUG 4+	98	S	
10028	B	B	2015	Communication	Telecommunicati	Business Applicati	Customer relations	Enhancement	Siebel	IFPUG 4+	138	M1	
10046	B	B	2015	Communication	Telecommunicati	Business Applicati	Customer relations	Enhancement	Java	IFPUG 4+	63	S	
10069	B	B	2013	Communication	Telecommunicati	Business Applicati	Data Warehouse	Enhancement	Shell	IFPUG 4+	949	M2	
10085	A	B	2010	Electronics & Com	Computers & Software;			Enhancement	ABAP	IFPUG 4+	416	M2	
10109	B	B	2015	Insurance	Insurance;	Business Applicati	Workflow support	New Developme	.Net	NESMA	317	M2	
10132	B	B	2010	Manufacturing	Manufacturing;	Business Applicati	Cars selling;	Enhancement	Java	IFPUG 4+	260	M1	
10145	B	B	2011	Manufacturing	Manufacturing;	Mathematically-Int	Computing of the th	Enhancement	MATLAB	IFPUG 4+	99	S	
10149	B	B	2012	Communication	Telecommunicati	Mathematically-Int	Customer relations	Enhancement	PL/SQL	IFPUG 4+	244	M1	
10169	B	B	2015	Insurance	Insurance;	Business Applicati	Workflow support	Enhancement	Oracle	NESMA	168	M1	
10238	B	B	2014	Communication	Telecommunicati	Business Applicati	Customer relations	Enhancement	Siebel	IFPUG 4+	284	M1	
10248	A	A	2012	Communication	Telecommunicati	Business Applicati	Online analysis an	Enhancement	Java	IFPUG 4+	302	M2	
10305	B	B	2015	Communication	Telecommunicati	Business Applicati	Customer relations	Enhancement	Java	IFPUG 4+	110	M1	
10313	B	B	2015	Insurance	Insurance;	Business Applicati	Workflow support	Enhancement	COBOL	NESMA	402	M2	
10317	B	B	2015	Government	Government;	Business Applicati	Business Applicati	Enhancement	.Net	NESMA	8	XXS	
10396	B	B	2014	Communication	Telecommunicati	Business Applicati	Customer relations	Enhancement	Visual Basic	IFPUG 4+	165	M1	
10404	B	B	2010	Manufacturing	Manufacturing;	Business Applicati	Logistic indicators;	Enhancement	Java	IFPUG 4+	72	S	
10412	B	B	2013	Insurance	Insurance;	Business Applicati	Unknown;	Enhancement	PL/SQL	IFPUG 4+	53	S	

## EXAMPLE

- Data Quality: A or B
- Year of Project > 2012
- Project Type: Enhancement
- Primary Programming language: Java
- Count approach: Nesma or IFPUG
  
- Further refinement, for instance:
  - Size category
  - Methodology
  - Industry
  - Application type
  - Team size
  - Time pressure (duration)
  - ...

Statistic	Hours/FP
Min	4,5
P25	5,8
Median	7,5
P75	8,3
Max	11,2
Average	7,8

Example: 500 FP Java project ROM Estimate

### Reality Zone:

Low (P25):	$1200 * 5,8 = 6.960$ hours
Likely(Median):	$1300 * 7,5 = 9.750$ hours
High (P75):	$1500 * 8,3 = 12.450$ hours

## EXAMPLE: HOW MANY FTE IN THE TEAM?

- Assuming a person (FTE) works 140 hours per month (varies per country), the team size necessary is:
  - Low:  $6.960 \text{ hours} / 6 \text{ months} / 140 \text{ hours per month} = 8,3 \text{ FTE}$
  - Average:  $9.750 \text{ hours} / 6 \text{ months} / 140 \text{ hours per month} = 11,6 \text{ FTE}$
  - High:  $12.450 \text{ hours} / 6 \text{ months} / 140 \text{ hours per month} = 14,8 \text{ FTE}$
- High-Level FPA in combination with ISBSG data is really useful for these type of estimation processes.
- When measuring the actual productivity sprint after sprint, in hours per FP, the actual velocity (delivery speed) in FP per month can be measured. When necessary early corrections can be made.

## CONCLUSION

- The IT world is moving towards agile teams, resulting in more agile metrics and declining use of function point metrics.
- Agile teams use story point on the team level, which is good. But management often tries to use story point metrics for management processes, which is impossible.
- Functional size measurement can easily be carried out on agile user stories, but few organizations do this.
- Functional size measurement in combination with ISBSG data is an invaluable source of information for many management processes, like:
  - (long-term) estimation of agile teams: How much functionality will be ready in X months with a team size of Y people
  - Supplier contracting (price per function point, instead of Time and Material)
  - (Supplier) Performance Measurement
  - Progress tracking and forecasting
  - Budgeting,
  - Benchmarking
  - ...



## CRY OUT TO THE INDUSTRY!



Use **Story points** on the team level



25	5,
Median	7,5
P75	8,3

A circular inset showing a table with statistical data. The table has two columns and three rows. The first row contains 25 and 5,. The second row contains Median and 7,5. The third row contains P75 and 8,3.

use **function points** and **historical data** to support decision making in management processes !!

**THANK YOU!**



[www.linkedin.com/in/haroldvanheeringen](http://www.linkedin.com/in/haroldvanheeringen)



@haroldveendam



haroldveendam

ISBSG: [www.isbsg.org](http://www.isbsg.org)  
Nesma: [www.nesma.org](http://www.nesma.org)  
METRI: [www.metrigroup.com](http://www.metrigroup.com)



harold.van.heeringen@metrigroup.com  
President@isbsg.org