ISBSG
Delivering IT Confidence

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

SOFTWARE COST ESTIMATION

FINALLY A REAL PROFESSION !

IT Confidence 2018, Mexico City

12th September 2018

Harold van Heeringen, ISBSG president

INTRODUCING ME

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

President@isbsg.org

www.linkedin.com/in/haroldvanheeringen

@haroldveendam

haroldveendam

ISBSG: www.isbsg.org
Nesma: www.nesma.org
METRI: www.metrigroup.com

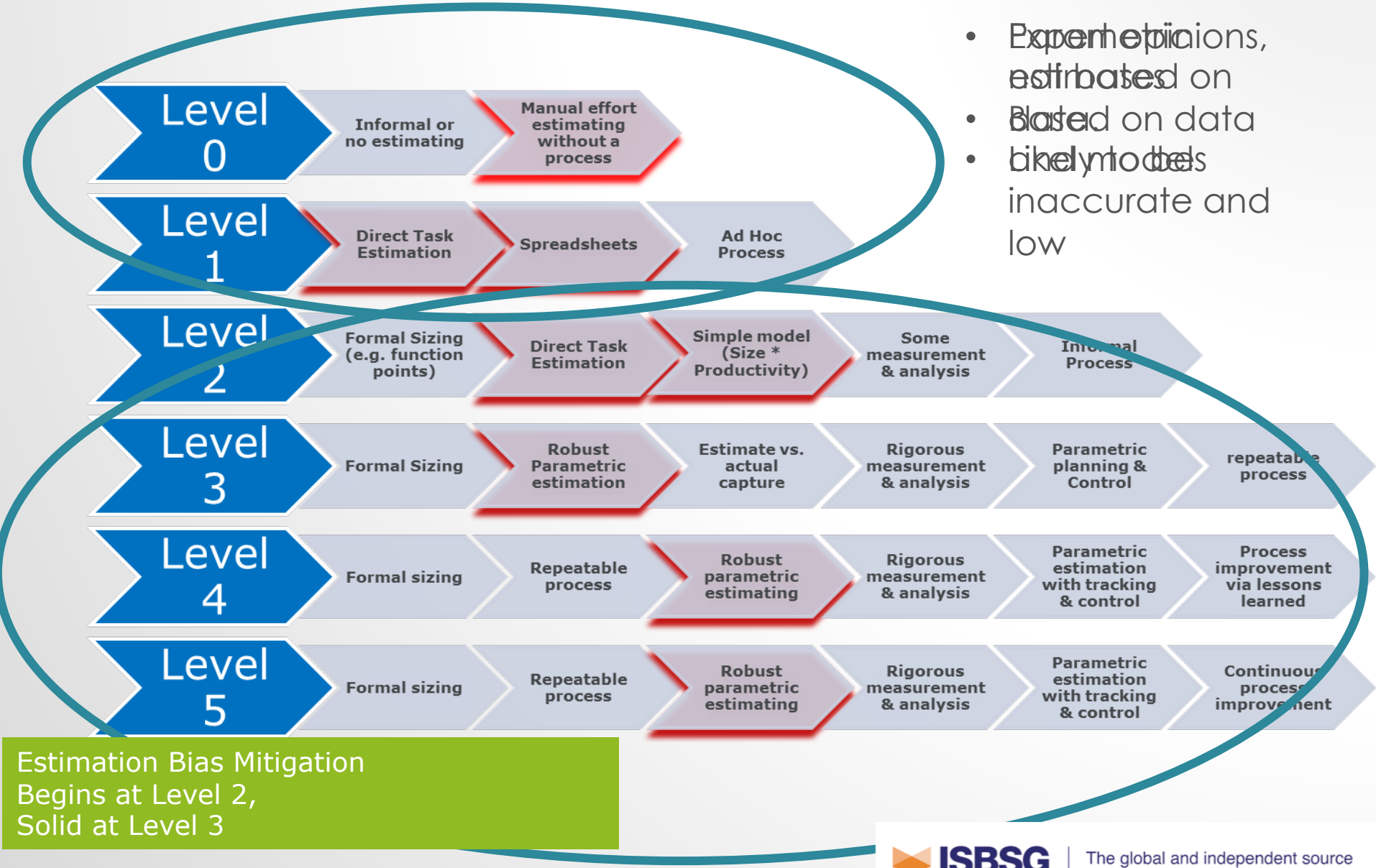
TOPICS ADDRESSED

- Industry Maturity
- Software Estimation – the current practice
- Software Cost Estimation basics
- Introducing the Software Cost Estimation Body of Knowledge
- Next steps
- Conclusions.

COST ESTIMATE

- A **cost estimate** is the approximation of the cost of a program, project, or operation. The cost estimate is the product of the cost estimating process. The cost estimate has a single total value and may have identifiable component values.
- A **problem with a cost overrun** can be avoided with a credible, reliable, and accurate cost estimate.
- A cost estimator is **the professional who prepares cost estimates**. There are different types of cost estimators, whose title may be preceded by a modifier, such as building estimator, or electrical estimator, or chief estimator. Other professionals such as quantity surveyors and cost engineers may also prepare cost estimates or contribute to cost estimates.
- In the US, according to the Bureau of Labor Statistics, there were **185400 cost estimators** in 2010.[1] There are around 75000 professional quantity surveyors working in the UK.
- Cost Estimator is **not a recognized profession** in the Software industry!
- Software Cost Estimates are created by project leaders, architects, developers, testers and other team members.
- Cost Estimates are typically based on Work Breakdown Structures, filled in with estimated hours based on 'experience' and 'gut feeling'. These are human 'expert' estimates.

ESTIMATING MATURITY MODEL*



- Expert opinions, estimates based on data
- Based on data
- Likely models inaccurate and low

Estimation Bias Mitigation Begins at Level 2, Solid at Level 3

* Developed by Dan Galorath, www.galorath.com

ESTIMATION IN THE SOFTWARE INDUSTRY

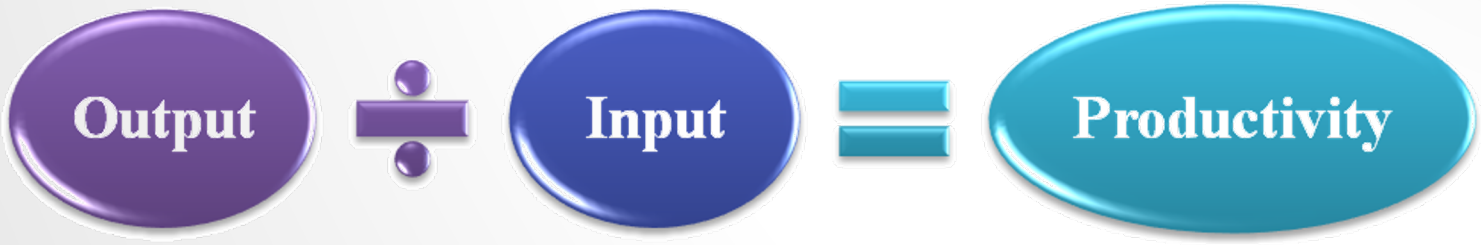
Software industry: low maturity in performance measurement and estimation

- 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 software development and maintenance 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

- But Software Development is becoming more and more important for organizations as delivering new software functionality fast becomes more and more a **driver for business success.**
- **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. hour 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**.
  - **Input** is usually easily measured in: 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?

What is Productivity?



Productivity is expressed or measured as a mathematical division of following parameters (functions).

Output / Input = Productivity

Example: Painting a wall: Productivity is X m^2 per hour

OUTPUT MEASUREMENT

Input is often easy to measure: X hours spent on release X

Output is usually much harder to measure. What was delivered in release X?

- The size of the delivered 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, many organizations don't have the knowledge, expertise and skills to do this. Therefore the output is not measured at all, or measured in a non-standardized way, resulting in low maturity.
- **As output is often not measured in a standardized way, there is little available data about productivity. This results in low estimation maturity as well.**

SOFTWARE SIZE

- How ‘big’ is the software to be developed, or maintained?
- But 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.

Most recently:

- Story Points - **Not standardized. Not Objective. Not repeatable. Not a measure of size!**

- **Only ISO/IEC methods for functional size measurement are applicable:**
  - Nesma function points – **ISO/IEC 24570**
  - COSMIC function points – **ISO/IEC 19761**
  - IFPUG function points – **ISO/IEC 20926**

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^2 glass wall = 20 m^2 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: **more value!**
- Functional size is the basis for **objective software metrics**:
  - Productivity (Hours spend per FP)
  - Cost Efficiency (Money spend per FP)
  - Time to Market (FP per calendar month)
  - Quality (Defects per 1000 FP)

REALISTIC ESTIMATION IS KEY

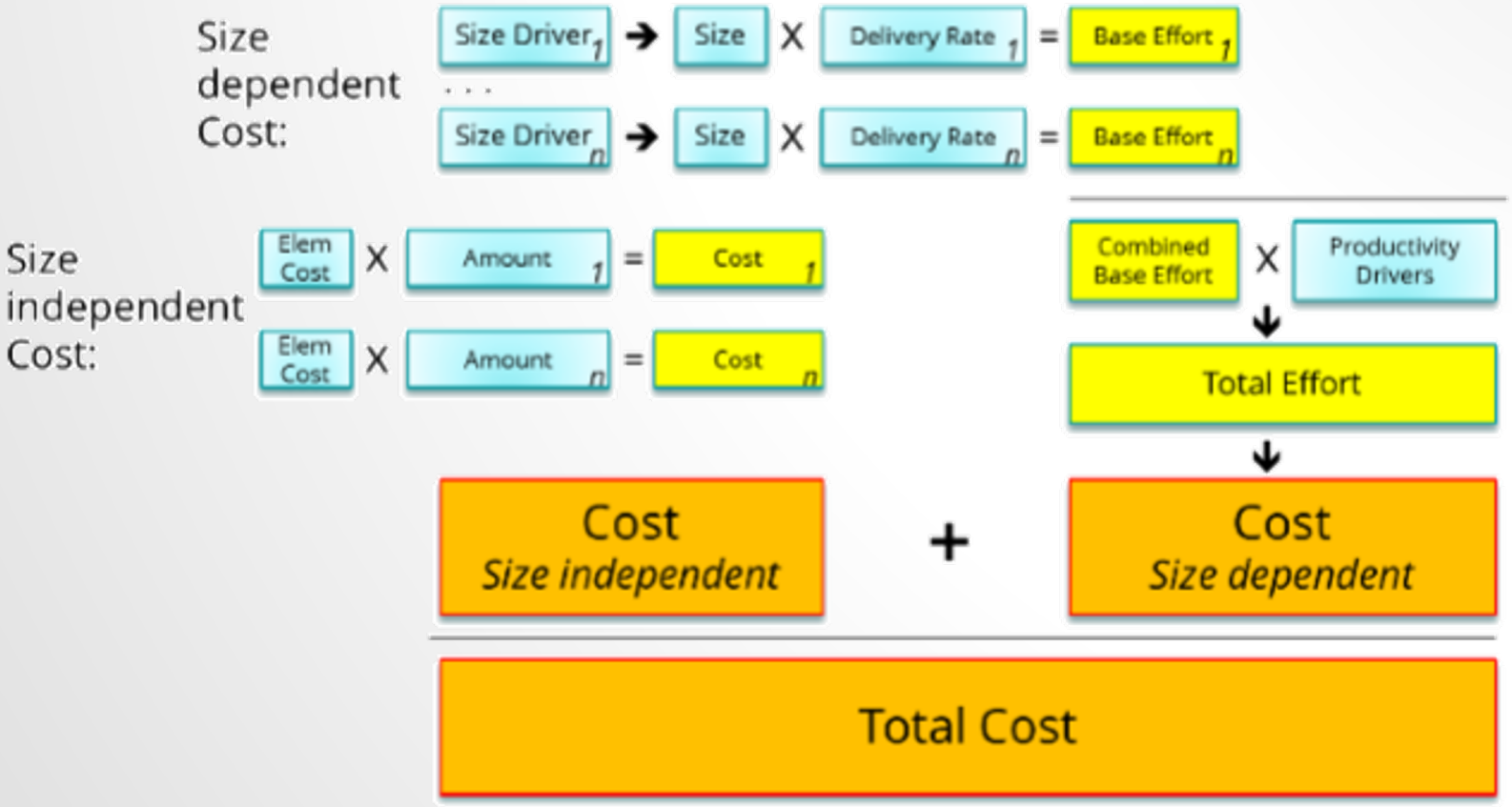
A realistic estimate is one of the most important **conditions** for a successful project.

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
- The feeling of the team and the stakeholders

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

ESTIMATE BREAKDOWN



AVOID OPTIMISM!!

- **Many projects are not estimated in a professional way**
  - Only expert estimates, no use of estimation models / historical data

- **Underestimation results in bad planning**
  - Development team too small
  - Duration too short
  - Unrealistic milestones
  - Project management with no grip on the project
  - Extra management attention, more meetings
  - Stress in the team → bad quality → more effort
  - Bad software, low maintainability

IN THE IT INDUSTRY, ESTIMATES ARE OFTEN LOW

IT industry – estimates are too optimistic

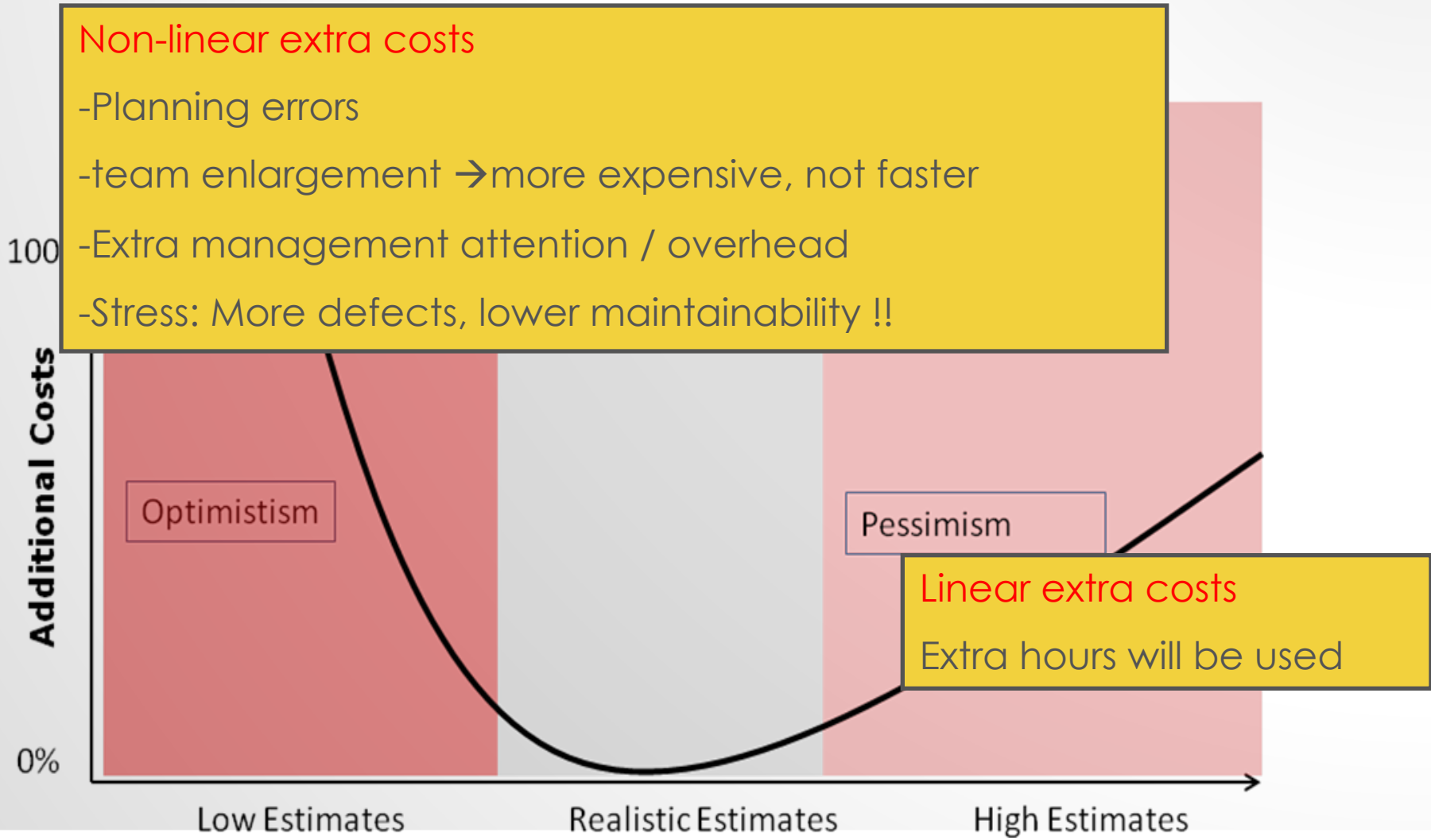
- Business/customer: pressure to lower price
- Business/customer: pressure shorter time-to-market
- Business/customer: incomplete requirements
- Business/customer: early fixed price/date quote

IT supplier

- Unclear what customer wants
- Immature estimation techniques (only expert estimates)
- No idea about own performance and capabilities;
- Not defendable → easy to push back

Optimistic estimates are more rule than exception

REALISTIC ESTIMATES ARE IMPORTANT!



TWO WAYS TO ESTIMATE

Level 3, 4 and 5

Objective

Size

nesma

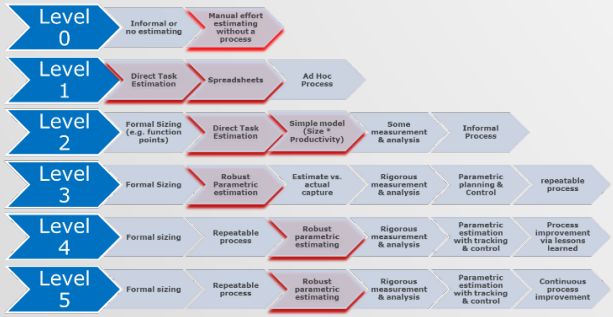
Estimating &

Benchmarking

Effort

Cost

Level 0, 1 and 2



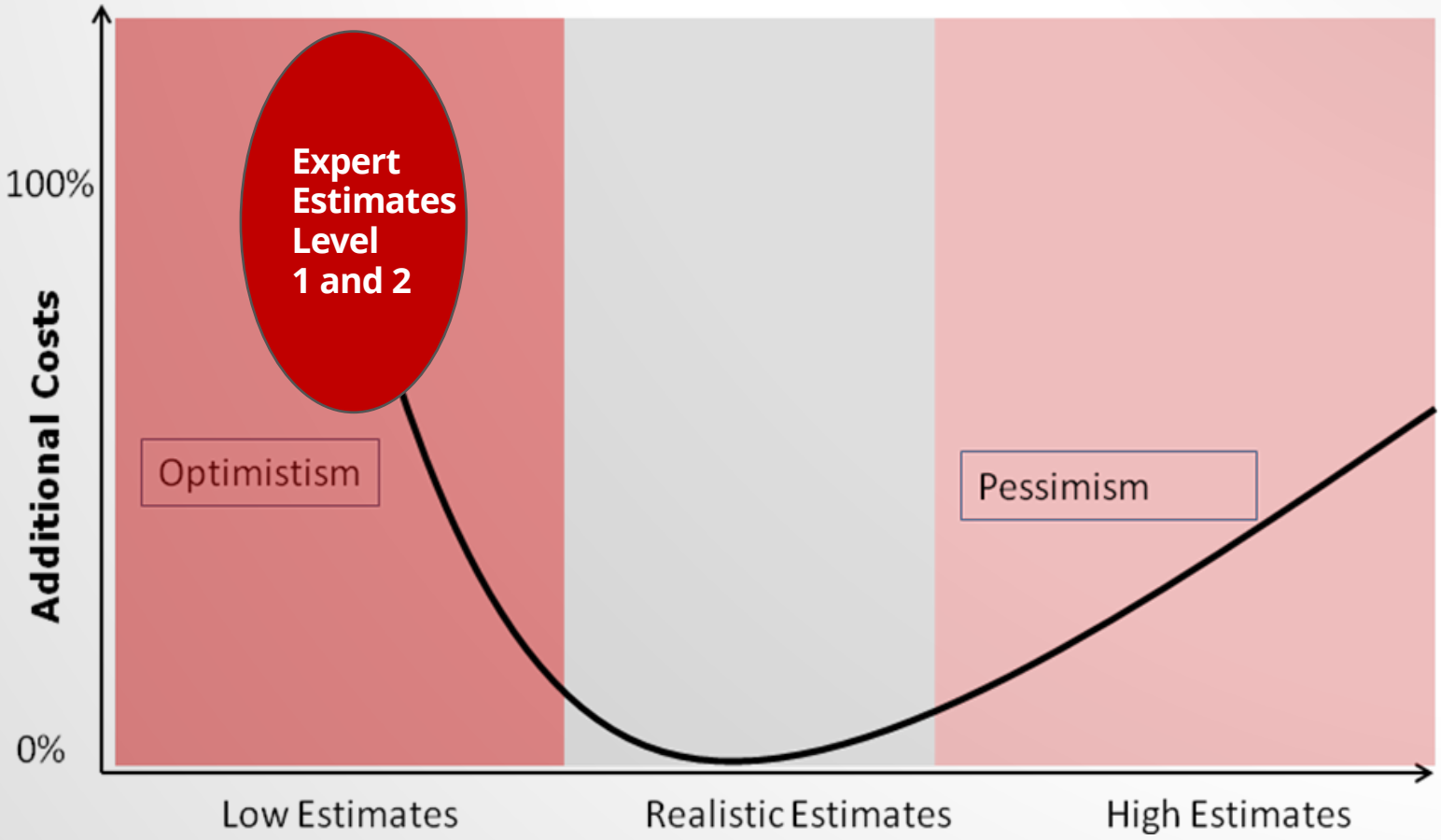
LEVEL 1 AND 2 ESTIMATES

- Bottom-up , assign effort hours to work items, based on knowledge and experience

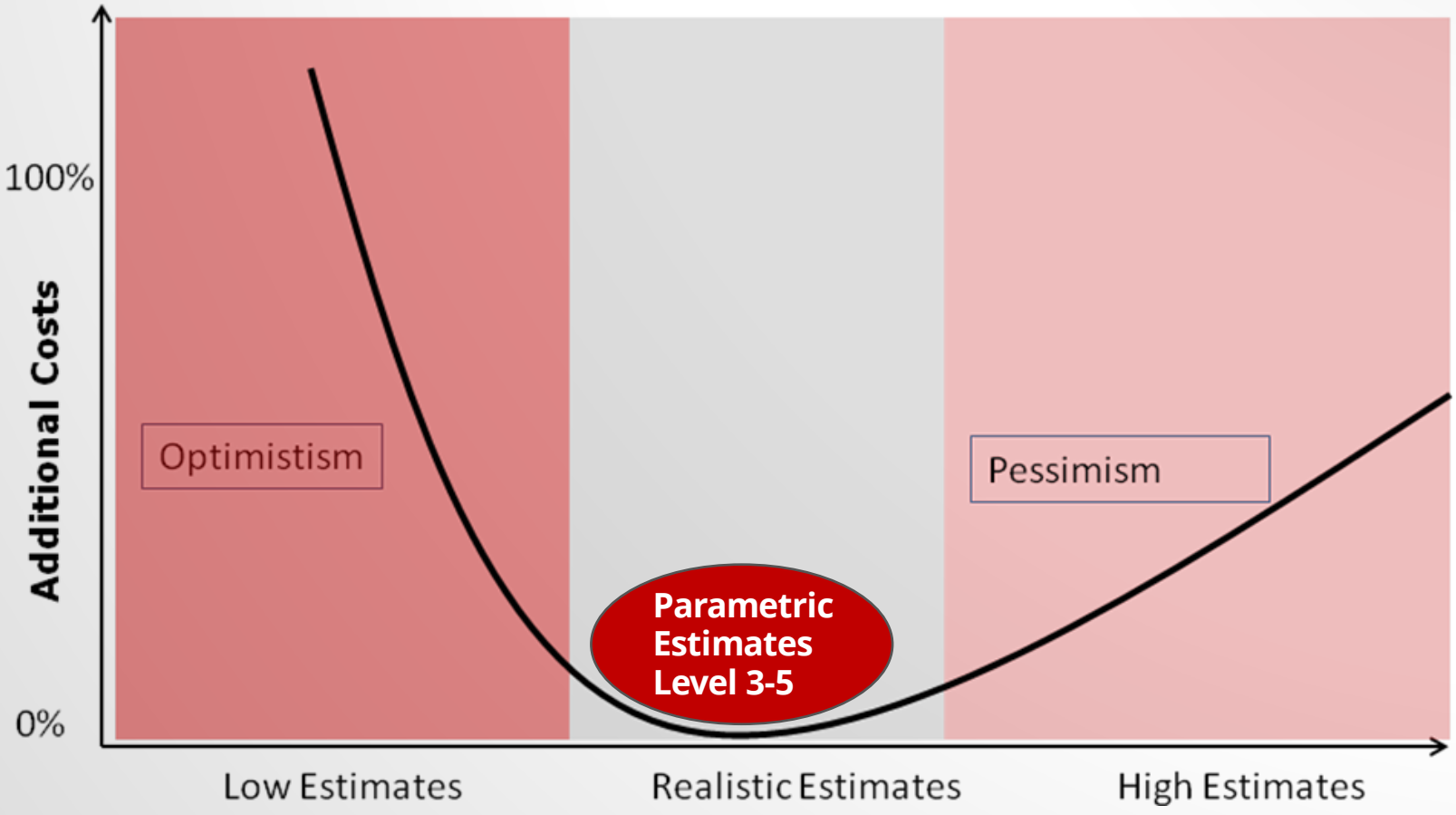
Result: expert estimates are optimistic, on average 30% underestimation.

- **Disadvantages:**
  - Forgotten activities (e.g. test script reviews. …);
  - No good foundation of the estimate, very subjective;
  - The expert is not going to de all the work (who will ?);
  - How expert is the expert? (projects are unique);
  - Experts don't take into account duration, team size, etc.;
  - Experts don't assess the reality value, no real use of history.

LEVEL O,1 AND 2 ESTIMATES: OPTIMISTIC AND FAILURE

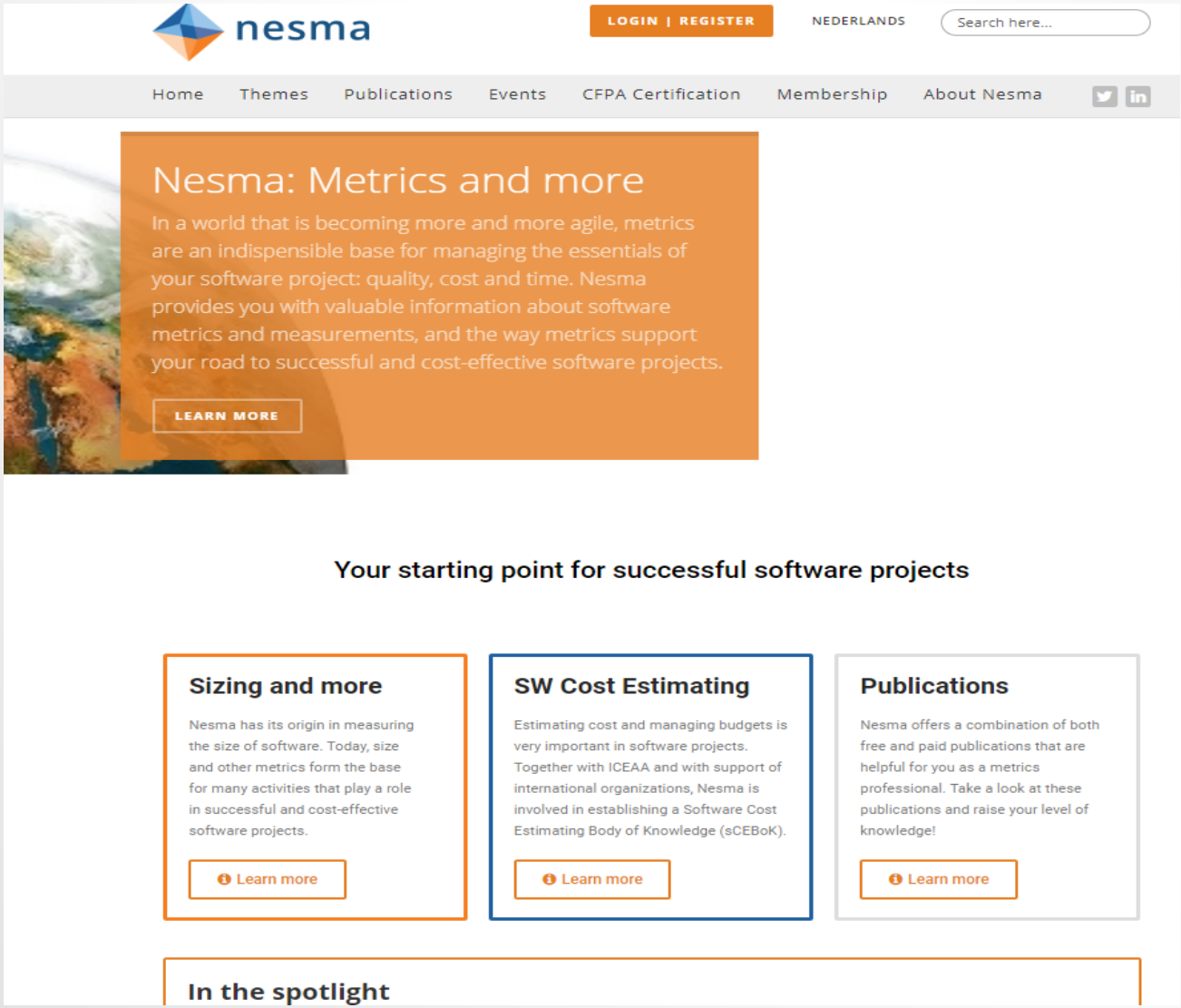


LEVEL 3-5 ESTIMATES: REALISTIC AND SUCCESS



ICEAA AND NESMA – SOFTWARE COST ESTIMATION BODY OF KNOWLEDGE

ICEAA AND NESMA – SOFTWARE COST ESTIMATION BODY OF KNOWLEDGE



SOFTWARE COST ESTIMATOR AS A PROFESSION

- Software Cost Estimating is the **profession** for estimating the costs of a software solution based on the needs and a defined solution.
- Software Cost estimation is the basis for a successful software project and needs to have the right level of detail and accuracy to be able to control the delivery with respect to scope, cost and schedule.
- A **certified software cost estimator** creates estimates based upon relevant historical data and by means of estimation models. Except cost, these estimates result in a schedule to deliver the defined solution.
- A certified cost estimator baselines his estimate in a **Basis of Estimate** document, documenting the scope, the assumptions, the data used and the models used per estimated item.

SOFTWARE COST ESTIMATOR CERTIFICATION

- Currently there is **no internationally accepted certification** for the profession of Software Cost Estimator available in the market, like the certification for Cost Estimator and Analyst for system engineering that is provided by ICEAA.
- To fill this gap, Nesma and ICEAA (International Cost Estimation and Analysis Association) are working together on the creation of a training curriculum and certification program to make 'Software Cost Estimator' a profession.
- Certified Software Cost Estimators will support organizations to bring their cost estimation process to a higher level and to make software projects more successful.
- The following certifications are anticipated:
  - Certified Software Cost Estimator Foundation
  - Certified Software Cost Estimator Expert

SOFTWARE COST ESTIMATION BODY OF KNOWLEDGE (SCEBOK)

- Software Cost Estimation Body of Knowledge (sCEBoK).
- This body of knowledge will contain relevant knowledge to fulfil the role of Software Cost Estimator and will be the basis for training and certification. Except knowledge, experience is an important criterion to pass this exam.

sCEBoK 2018 content

Currently the sCEBoK consist of the following modules, as presented during the ICEAA Conference in Phoenix in June 2018:

- Estimation principles
- Solution based estimation
- Basis of Estimate (BOE)
- Basis of Measurement (BOM)
- Budget process
- Data collection and basic data analysis
- Statistics to support basic metric analysis
- Estimation in the software lifecycle
- Estimation methods – Formal
- Estimation models – Size based methods
- Estimation maturity
- Cost drivers
- Benchmarking
- Application Maintenance

For the final sCEBoK, additional modules will be developed and current modules will be further improved with support of professionals of international software organizations. The new modules will focus on knowledge with respect to software cost estimation as well as on a further refinement of estimation models for modern lifecycles like Agile and DevOps. The sCEBoK will consist of training material including detailed notes. As a next step a wiki will be developed to share the knowledge.

INTERNATIONAL SOFTWARE BENCHMARKING STANDARDS GROUP (ISBSG)

- Independent and not-for-profit
- Full Members are non-profit organizations, like Nesma, IFPUG, FiSMA, China SPI, GUFPI-ISMA, JFPUG, Swiss-ICT
- Grows and exploits two repositories of software data:
  - New development projects and enhancements (> 8250 projects, releases and sprints)
  - Maintenance and support (> 1100 applications)

- Everybody can submit project data
  - DCQ's on the site, online or Excel data files
  - Anonymous
  - Free benchmark report in return

ISBSG MISSION

- Mission: "**To improve the management of IT resources** by both business and government, through the provision and exploitation **of public repositories of software engineering knowledge** that are standardized, verified, recent and representative of current technologies"

- All ISBSG data is
  - validated and rated in accordance with its quality guidelines
  - current
  - representative of the industry
  - independent and trusted
  - captured from a range of organization sizes and industries

ISBSG DATA

>8000 rows in Excel, Easy to analyze.

>250 data fields (columns) per project

ISBSG Delivering IT Confidence

D&E Release April 2016

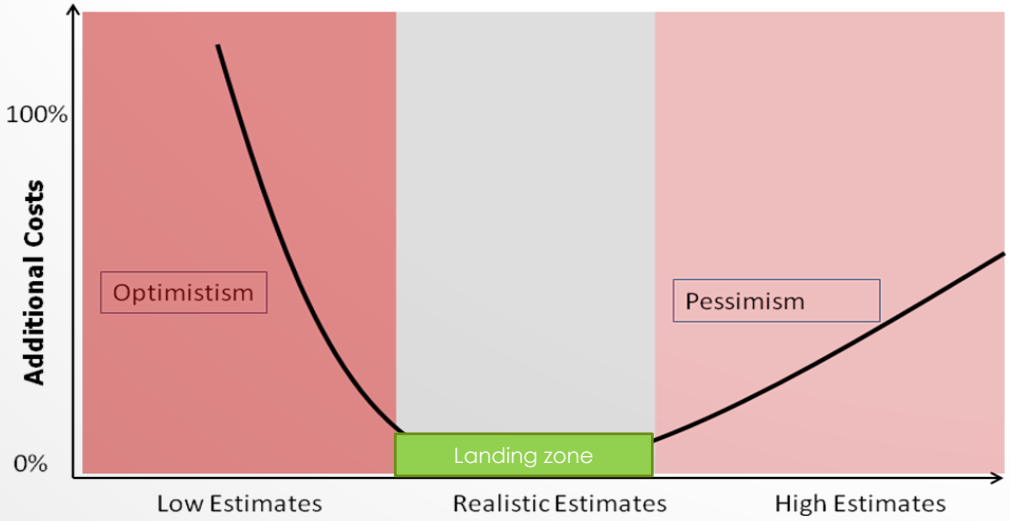
7518 rows

	Rating	*Rating*	*Software Age*	*Major Grouping*	*Major Grouping*	*Major Grouping*	*Major Grouping*	*Major Grouping*	*Major Grouping*	*Major Grouping*	*Major Grouping*	*Major Grouping*
ISBSG Project ID	Data Quality Rating	UFP rating	Year of Project	Industry Sector	Organisation Type	Application Group	Application Type	Development Type	Development Platform	Language Type	Primary Programming Language	Count Approach
10046	B	B	2015	Communication	Telecommunications	Business Applicatio	Customer relationsh	Enhancement	Multi	3GL	Java	IFPUG 4+
10109	B	B	2015	Insurance	Insurance;	Business Applicatio	Workflow support &	New Developmen	PC	4GL	.Net	NESMA
10169	B	B	2015	Insurance	Insurance;	Business Applicatio	Workflow support &	Enhancement	PC	4GL	Oracle	NESMA
10305	B	B	2015	Communication	Telecommunications	Business Applicatio	Customer relationsh	Enhancement	Multi	3GL	Java	IFPUG 4+
10317	B	B	2015	Government	Government;	Business Applicatio	Business Applicatio	Enhancement		4GL	.Net	NESMA
10469	B	B	2015	Communication	Telecommunications	Business Applicatio	Stock control & orde	Enhancement	Multi	3GL	Java	IFPUG 4+
10665	B	B	2015	Communication	Telecommunications	Business Applicatio	Stock control & orde	Enhancement	Multi	3GL	Java	IFPUG 4+
10762	B	B	2015	Communication	Telecommunications	Business Applicatio	Customer relationsh	Enhancement	Multi	3GL	Java	IFPUG 4+
10940	B	B	2015	Insurance	Insurance;	Business Applicatio	Workflow support &	Enhancement	PC	3GL	Java	NESMA
11118	B	B	2015	Communication	Telecommunications	Business Applicatio	Logistic or supply pl	Enhancement	Multi	3GL	Java	IFPUG 4+
11230	B	B	2015	Insurance	Insurance;	Business Applicatio	Electronic Data Inte	Enhancement	PC	3GL	Java	NESMA
11318	B	B	2015	Communication	Telecommunications	Business Applicatio	GEO Information M	Enhancement	Multi	3GL	Java	IFPUG 4+
11737	B	B	2015	Communication	Telecommunications	Business Applicatio	Workflow support &	Enhancement	Multi	3GL	Java	IFPUG 4+
11990	B	B	2015	Insurance	Insurance;	Business Applicatio	Electronic Data Inte	New Developmen	PC	3GL	Java	NESMA
12928	B	B	2015	Insurance	Insurance;	Business Applicatio	Workflow support &	Enhancement	PC	4GL	.Net	NESMA
13120	B		2015	Service Industry	Art , Events , Ticketi	Business Applicatio	Document manager	Re-development	PC	3GL	Java	COSMIC
13137	B	B	2015	Communication	Telecommunications	Business Applicatio	Workflow support &	Enhancement	Multi	3GL	Java	IFPUG 4+
13372	B	B	2015	Insurance	Insurance;	Business Applicatio	Web-based Applica	New Developmen	PC	4GL	.Net	NESMA
14138	B	B	2015	Insurance	Insurance;	Business Applicatio	Workflow support &	Enhancement	PC	3GL	Java	NESMA

EXAMPLE ESTIMATE LANDING ZONE

- Selection:
  - Data Quality: A or B
  - Year of Project > 2012
  - Project Type: Enhancement
  - Primary Programming language: Java
  - Count approach: Nesma or IFPUG
- The landing zone may be in this case:
  - Low: 6.8 h/FP
  - Likely: 7.8 h/FP
  - Max: 9.4 h/FP
- Further refinement may be possible,
- for instance:
  - Size category
  - Development methodology
  - Industry
  - Application type
  - …

	PDR (hours/FP)
Number of projects	166
Minimum	4,2
Percentile 10%	5,3
Percentile 25%	6,8
Median	7,8
Percentile 75%	9,4
Percentile 90%	10,2
Maximum	15,3
Average	7,9



CONCLUSIONS

- Cost estimation is a profession in most industries, however Software Cost Estimation is not a profession yet as it is hard to measure software
- This is the root cause for many failing projects, which were estimated using low maturity techniques and therefore estimated optimistically
- There is loads of software estimation material, research and data available, but unknown to most of the industry
- ICEAA and Nesma create the Software Cost Estimation Body Of Knowledge (sCEBoK) and a certification program for certified Software cost estimators
- ISBSG is one of the contributing organizations, contributing effort and materials in order to improve estimation decision making in the industry.

THANK YOU!

www.linkedin.com/in/haroldvanheeringen

@haroldveendam

haroldveendam

ISBSG: www.isbsg.org
Nesma: www.nesma.org
METRI: www.metrigroup.com

President@isbsg.org