Estimating Test and Rework Effort in Software Projects

BrightX Online Events

Estimating Test and Rework Effort in Software Projects

Especially in complex projects, the challenges of inaccurate Estimation are in the Test and Rework effort.

Underestimating the effort in this area is mainly due to lack of a more mature approach to estimating the required test effort and the anticipated defects and thereby requiring rework effort. If all software projects use a Quantitative Approach for estimating Test Effort, Rework Effort, and anticipated Defects, the likelihood of successfully deliver the software project will increase. Using a quantitative approach will bring value and improvement to the estimated effort's accuracy, and the quality of the software product delivered.

This presentation will share the insight into Test and Rework Estimation and Estimation of anticipated Defects, using Function Point Analysis (FPA) and Software Non-functional Assessment Process (SNAP) – two industry sizing standards under the International Function Points Users Group (IFPUG).

Event: 4th of February 2021

Place: Virtual

Host Organization: Brightest

IP by Green

Estimating **Test** and **Rework** Effort in Software Projects

Christine Green

info@IPbyGreen.com

CHRISTINE GREEN

Independent Senior Analyst and Forecast Advisor

- 20+ years of experience in the software industry. Focus on complex and critical software projects, programs and contracts. Special focus area in forecast, scope analysis, contract price, cost and effort of delivery

- Worked on FPA contracts for both government and private sector since 2003 for EDS, HP (employed between 1996-2017) and as an independent consultant

- M.Sc. in Mathematics and Computer science

- Certified - PMP, CSM, SA, LSS BB & CFPS Fellow

www.linkedin.com/in/christinegreendk

President of IFPUG 2019-2021
CFPS Fellow 15th of September 2020
Certified CFPS since September 2000
Volunteered for IFPUG since 2003
Part of the CPM 4.3.1 review team
SNAP Project Manager until APM release 2.0

Interaction during the presentation

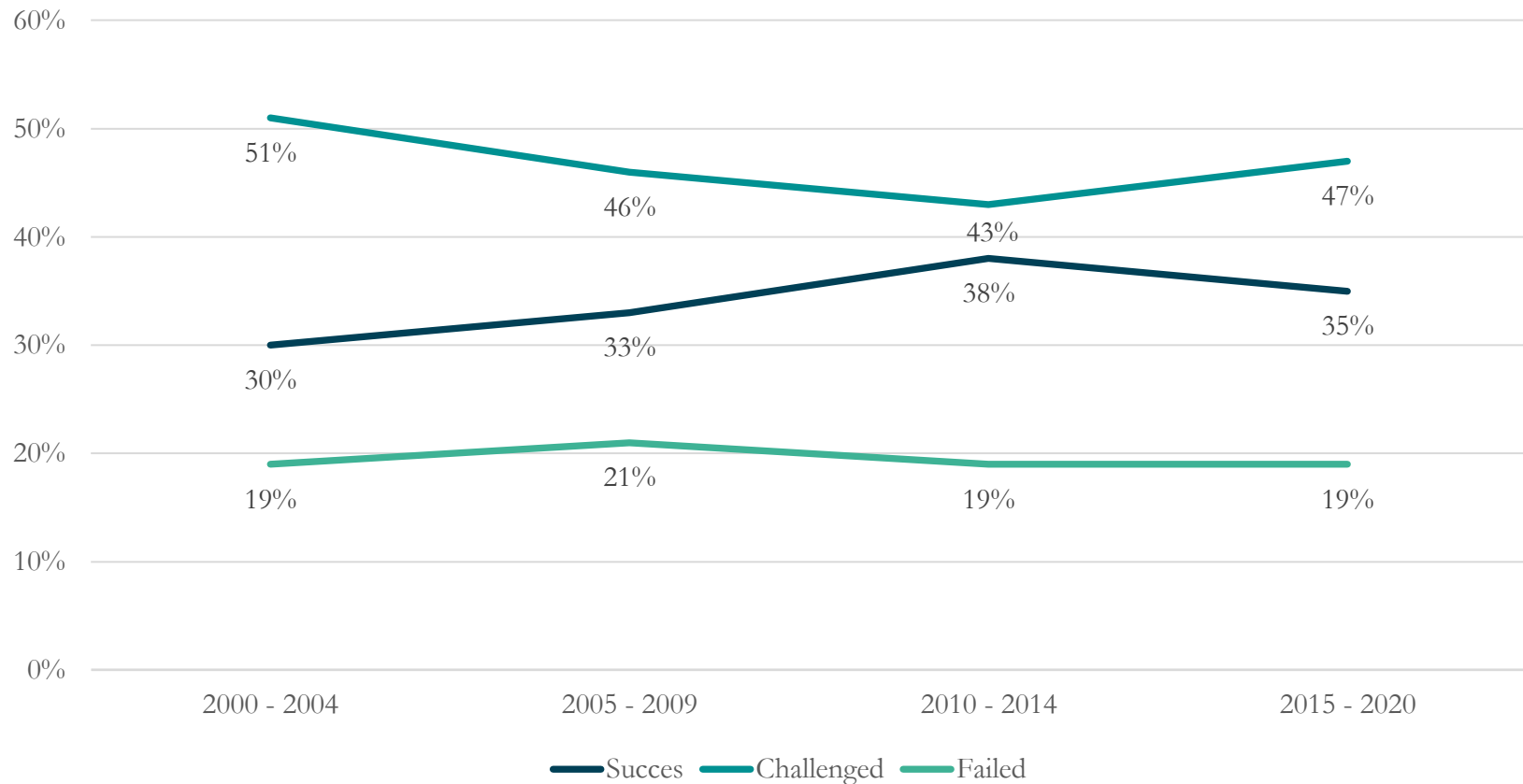https://ahaslides.com/BRIGHTEST1

IP by Green

Realistic Expectations

The single most important task of a project:
setting realistic expectations.

Unrealistic expectations based on inaccurate estimates are the single largest cause of software failure.

The improvement is missing

CHAOS 2020: Beyond Infinity

Project Success

There's no super-secret formula to project success.

Engaged executive sponsors

Projects aligned to organizational strategy
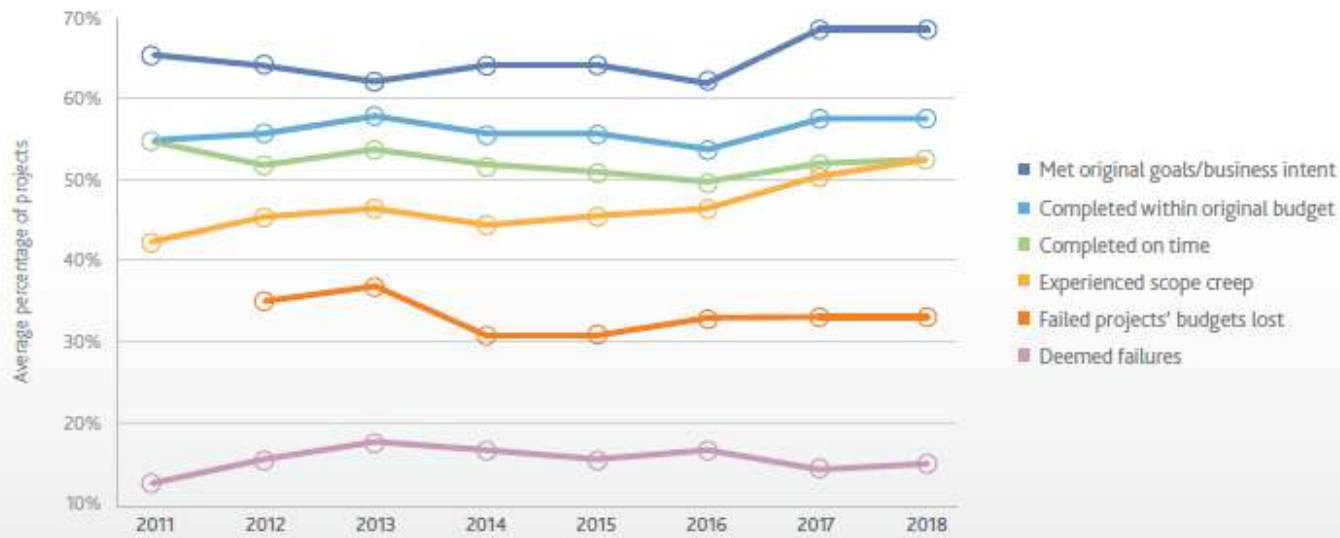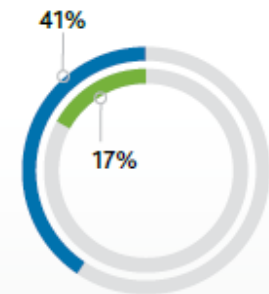
Control over scope creep

The Success of Scope Management

1. Investing in actively engaged executive sponsors

2. Avoiding scope creep or uncontrolled changes

3. Maturing value delivery capabilities

Inadequate Sponsor Support is a Primary Cause of Project Failure

41%

17%

■ **41% of Underperformers** say inadequate sponsor support is a primary cause of their failed projects

■ **17% of Champions** say inadequate sponsor support is a primary cause of their failed projects

Figure 6: Project Performance Metrics

Average percentage of projects

2011 2012 2013 2014 2015 2016 2017 2018

■ Met original goals/business intent
■ Completed within original budget
■ Completed on time
■ Experienced scope creep
■ Failed projects' budgets lost
■ Deemed failures

Applied Software Measurement

Agile and DevOps would shift left due to the move of critical testing practices earlier in the development lifecycle.
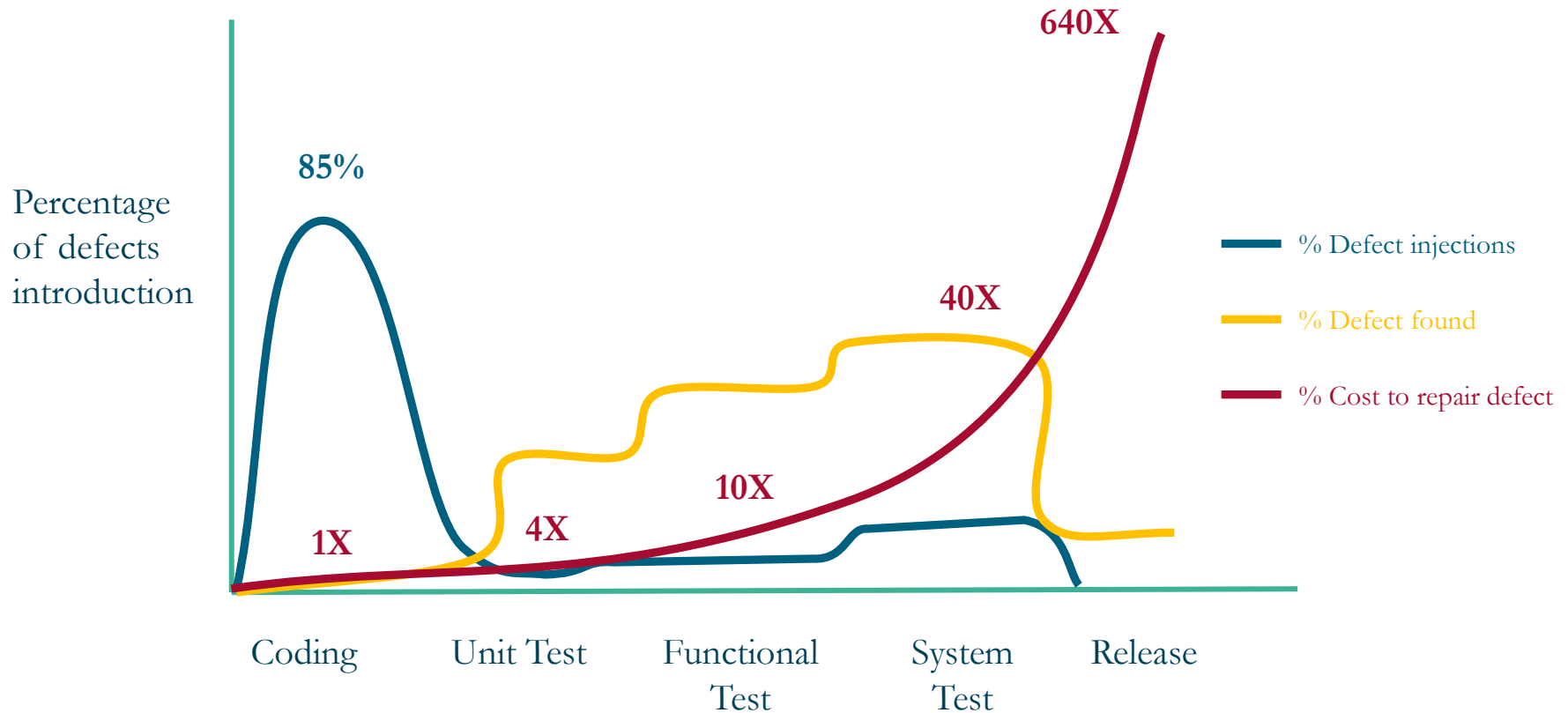
Phase Ratios – New Developments

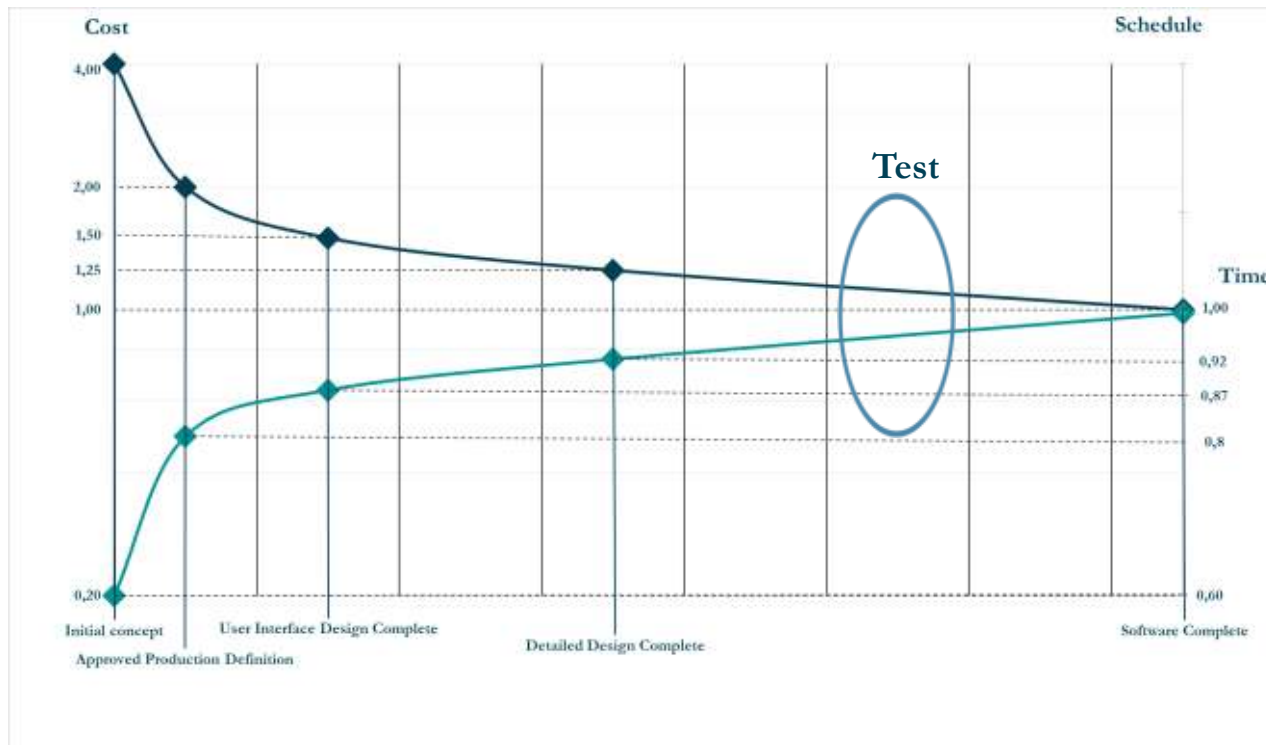| Phase Ratios-New Developments	
Phase	Ratios in pct.
Plan	9%
Specification	11%
Design	15%
Build	43%
Test	16%
Implementation	6%
Grand Total	100%
Source: ISBSG Benchmark R10	

Phase Ratios - Developments
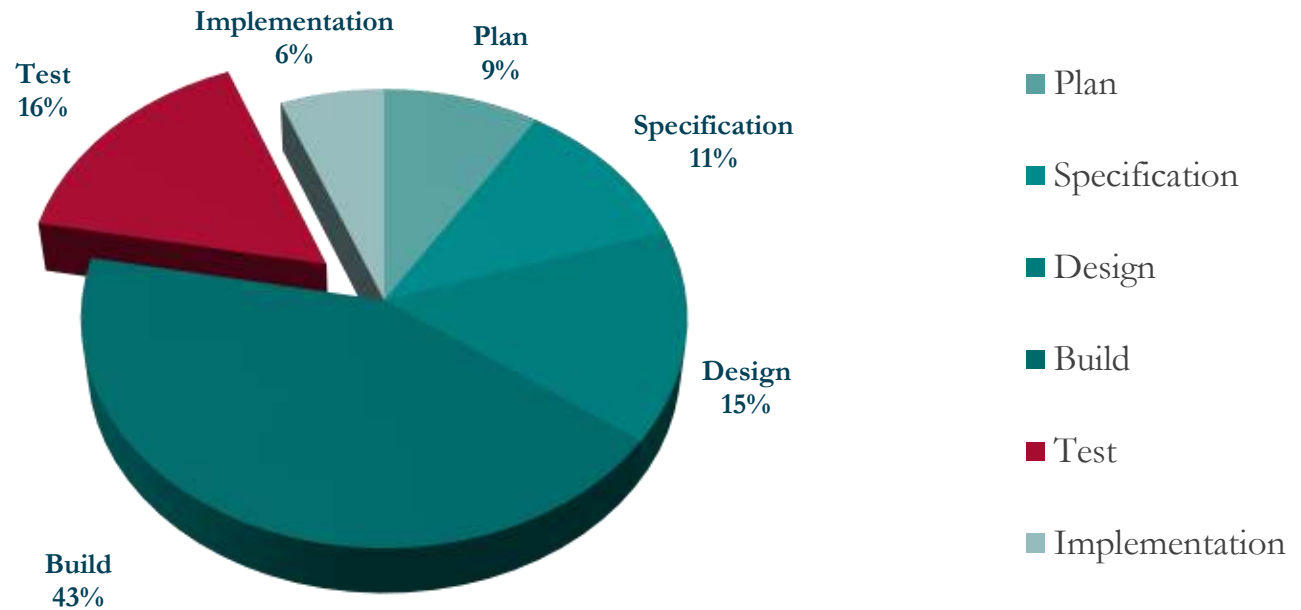
Test and Rework Estimation

- Plan, estimate and execute separately
  - Use metrics related inputs
    - Learn from the past

Example of focus on Testing

The Challenge

Target

Validation of estimates submitted by the two bidding IT providers...

» Estimates are in the ballpark

» Comparison with the market

» Forthcoming estimations: evaluation of an automated estimation tool (Quanter)

» Very tight deadline

» Lots of documentation

» Limited client support

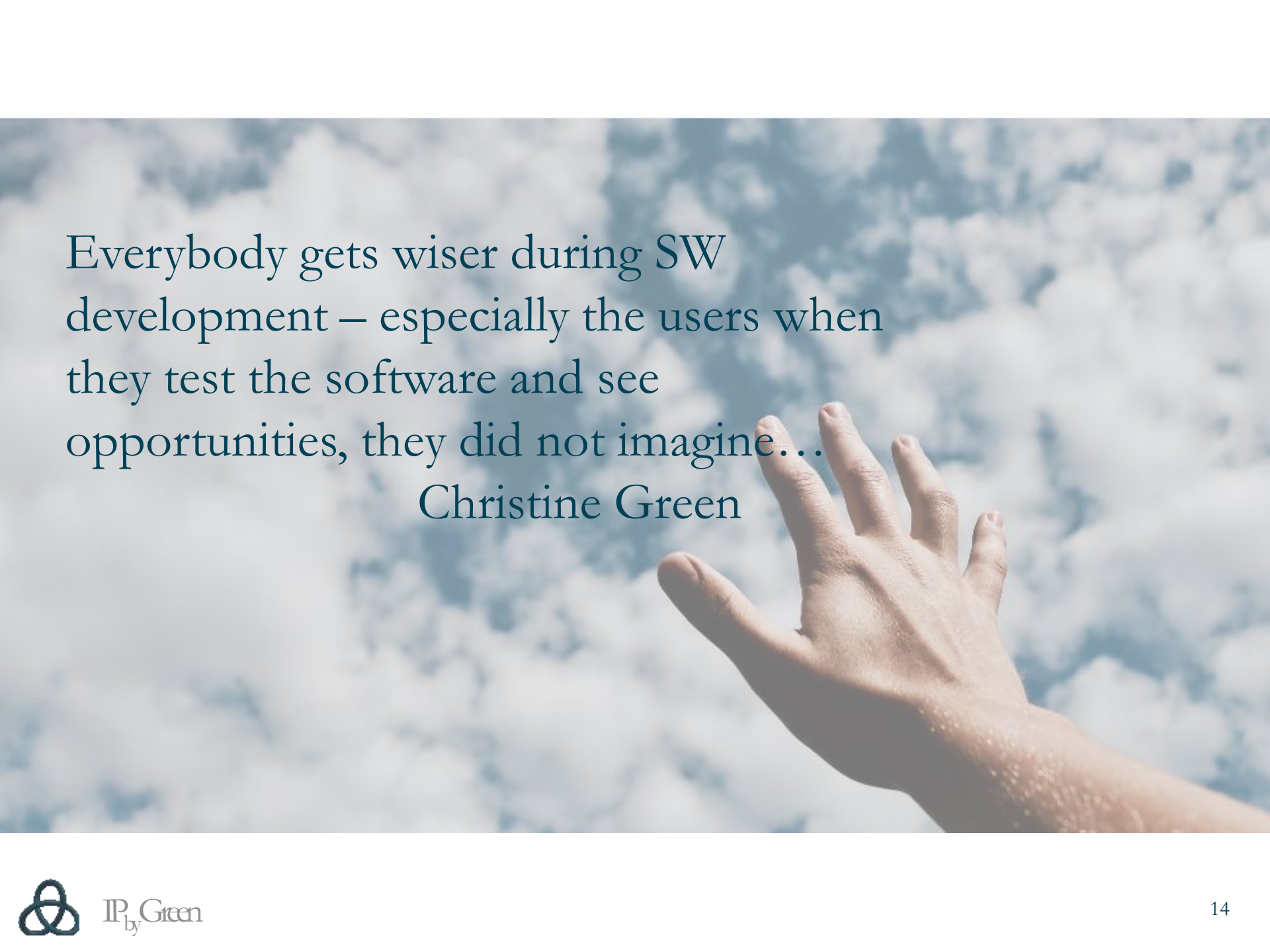» Release & Project Life Cycle oriented detail of estimated effort

BENEFITS FROM TESTING POINT OF VIEW

» To estimate the test cases needed to be design and execute.

» To estimate the effort, cost, people and time needed for the testing activities.

» Project planning considering testing activities and development team resources needed to solve the defects.

» Expected Quality and follow up of the actual project quality vs expected quality.

» Functional quality SLA for delivered projects in outsourcing contracts.
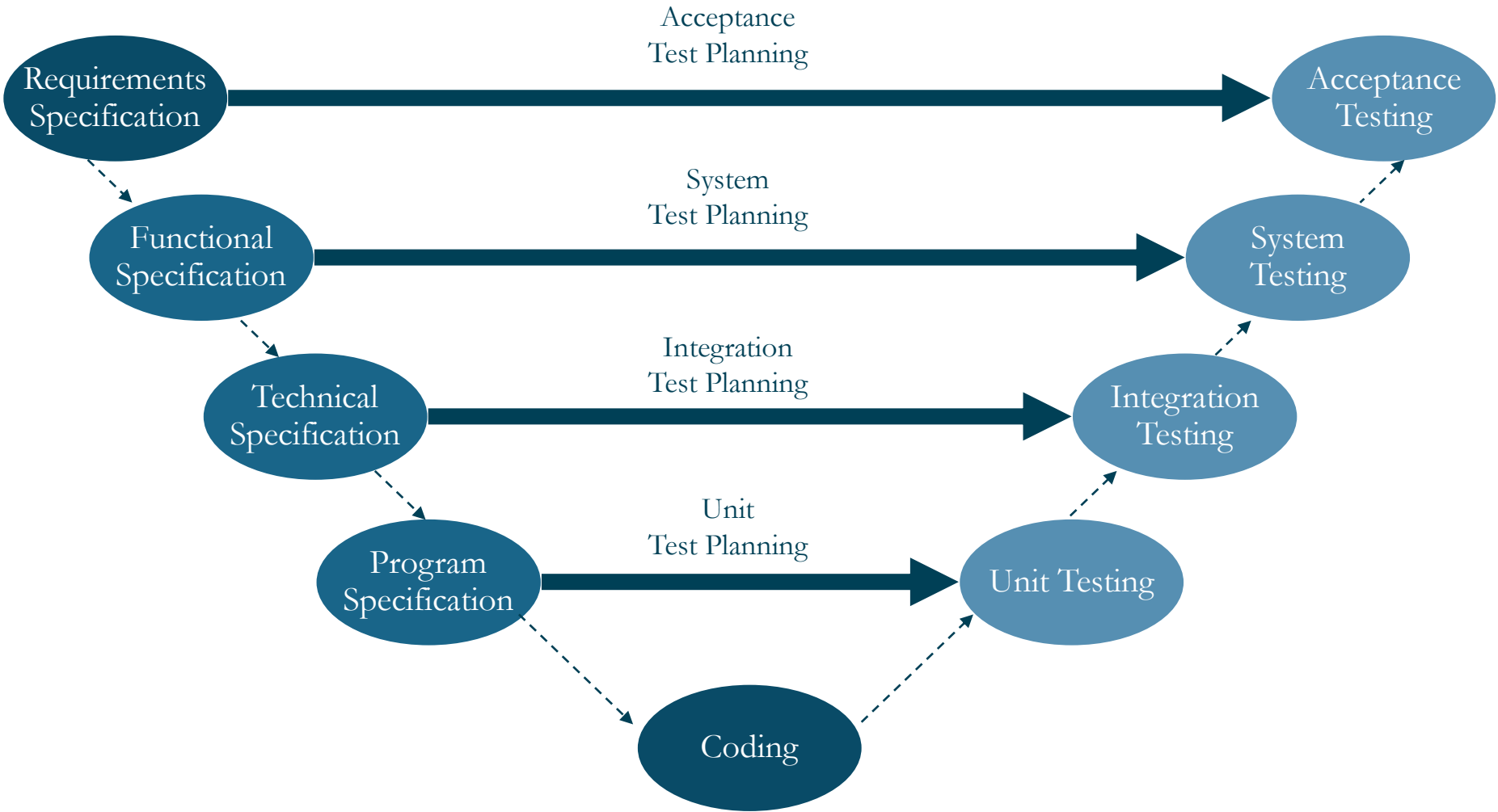
The 20M€ Tender Challenge Final **leda** MC

Everybody gets wiser during SW development – especially the users when they test the software and see opportunities, they did not imagine…
Christine Green

IP by Green

The V-model

Requirements Specification → Acceptance Testing (Acceptance Test Planning)

Functional Specification → System Testing (System Test Planning)

Technical Specification → Integration Testing (Integration Test Planning)

Program Specification → Unit Testing (Unit Test Planning)

Coding

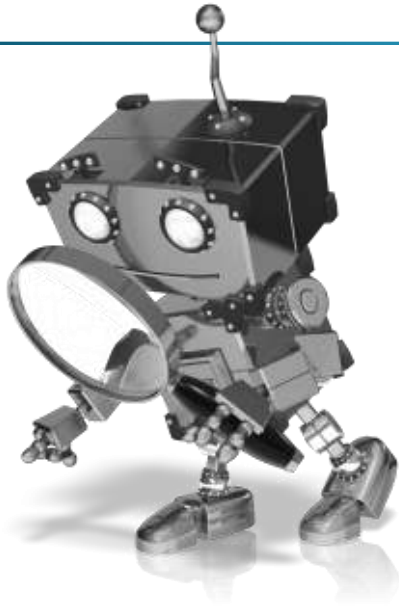IP by Green

Automated vs Manual Testing

Automated Estimate focus area:
- The definition and design
- The build of the automated test
- The test of the automated test

The rework
&
all that cannot be automated

Manual Estimate focus area:
- The definition and design
- The test execution
- The amount of resources

The rework
&
all the times you need to repeat the test

Measurements in Testing

Test effectiveness:

- Requirement and Design coverage

Test status:

- The number of tests run and their status
- The number of test cases and their status

Test resources:

- The time it takes to setup the Test and Defect Management Tool
- The time and cost it takes to plan, define
- The time it takes to "build"
- The time it takes to test and the number of times you need to repeat the test

Product quality:

- The number of defects and the rate of defect
- The time and cost it takes to fix - the Rework effort

Test Execution status

Passed
46%

Blocked
14%

■ Blocked

■ Failed
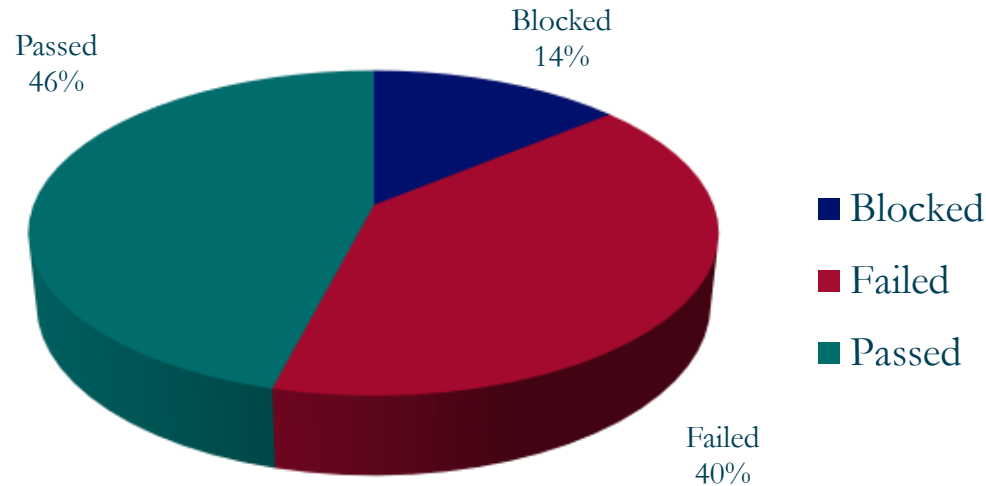
■ Passed

Failed
40%

Function
Point
Analysis

$$\text{pct. Test cases Blocked} = \frac{(\text{No. of Test cases Blocked}) * 100}{\text{Total no. of Test cases Executed}}$$

$$\text{pct. Test cases Failed} = \frac{(\text{No. of Test cases Failed}) * 100}{\text{Total no. of Test cases Executed}}$$

$$\text{pct. Test cases Passed} = \frac{(\text{No. of Test cases Passed}) * 100}{\text{Total no. of Test cases Executed}}$$

IP by Green

Function Point Analysis – the Method

FPA

Function Point Analysis

CONSISTENT AND STABLE SOFTWARE SIZE

A Certified FP Specialist is an expert in bridging between User, Technical and Planning needs

- ✓ **Business Process & Software Requirements Breakdown**
- • Scope Control from a Business perspective
- ✓ **Scope analysis and control from a user's perspective**
- • Scope of work for Teams and Projects
- • List of deliverables
- ✓ **List of transactions to be tested**
- • Measure of impact
- ✓ **Key Performance normalization factor**
- • The Functional Size Measure of a project or software component

The Process for scope illumination, control and measurement

Function Point Analysis
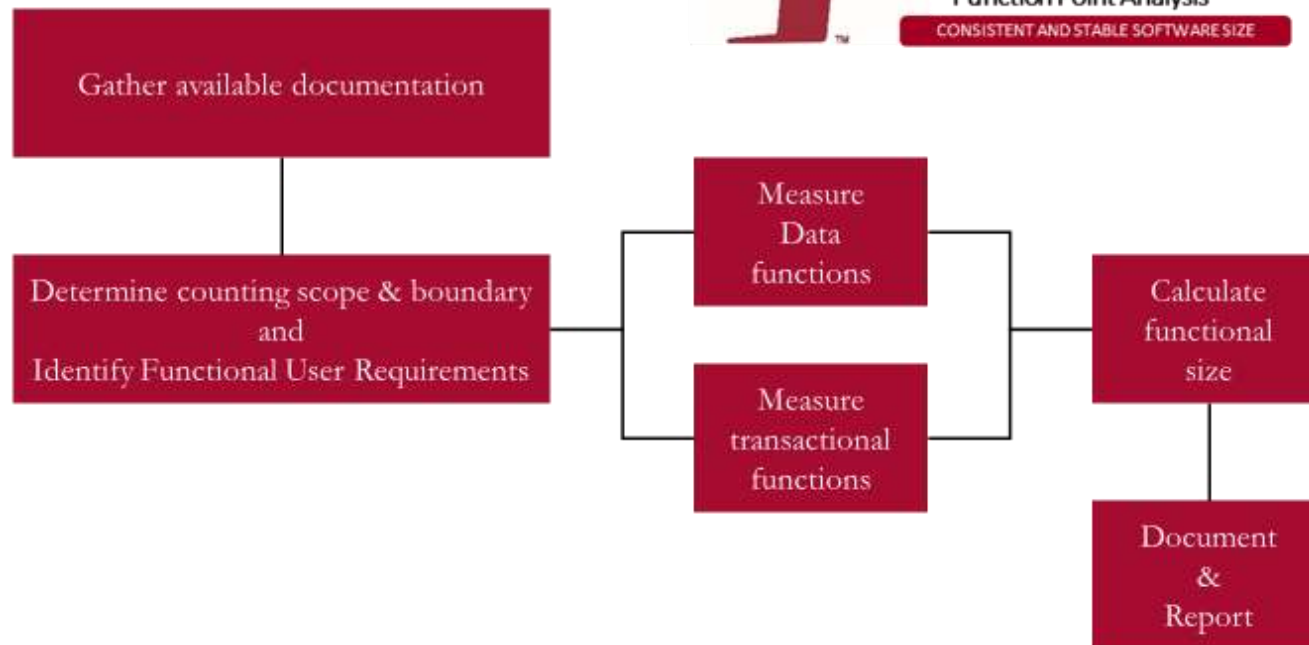
Join us at IFPUG.org and LinkedIn

Estimation and Monitoring

What does it take to build a LEGO construction?

Simple construction   Complex construction
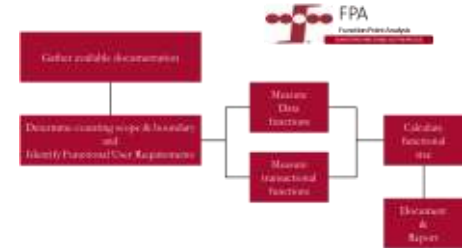
…it certainly depends on the requirements
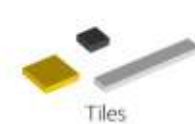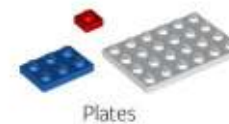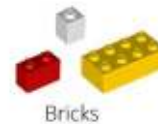
User View

User

- Any person or thing that communicates or interacts with the software at any time.

User recognizable

- Requirements for processes and/or data that are agreed upon, and understood by, both the user(s) and software developer(s).

User view

- Functional User Requirements as perceived by the user.

Meaningful.

- User recognizable and satisfies a functional requirement.

From the user view
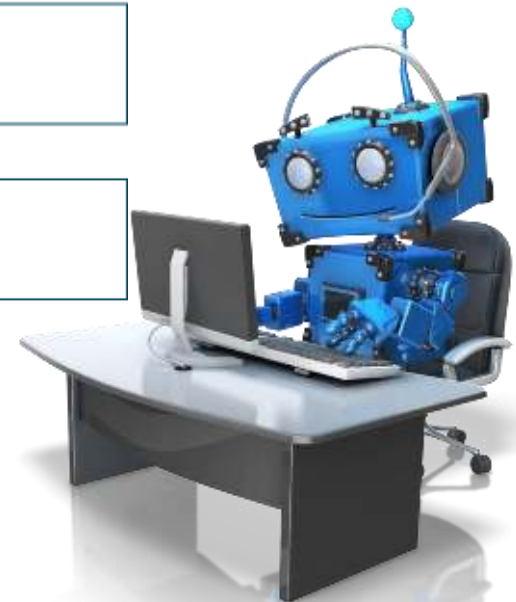
Boundaries from a business perspective

Business Process we refer to

External Boundary
Reference data
– not in scope except EIFs

External Boundary

Business Process where the scope includes perspective of stakeholders, engagement or data exchange

Important Interfaces

Business Process

Business Process

Important Reports

The Focus Boundary

The Focus boundary

Business Processes relevant for this boundary is identified. Do not consider how the business process is developed or delivered when considering boundaries.

Boundary Drawing
– The 1000 feet

Application C

Business Area D

Important Input

Application A

DB

Business Area A

Business Area B

Business Area C

Important Report

Report

Receiving and Sending

Application B

Business Area D

Report

Transactional Functions

	EI	EO	EQ
	Important Input – Maintain data	Reporting with processing logic	Find and View Data
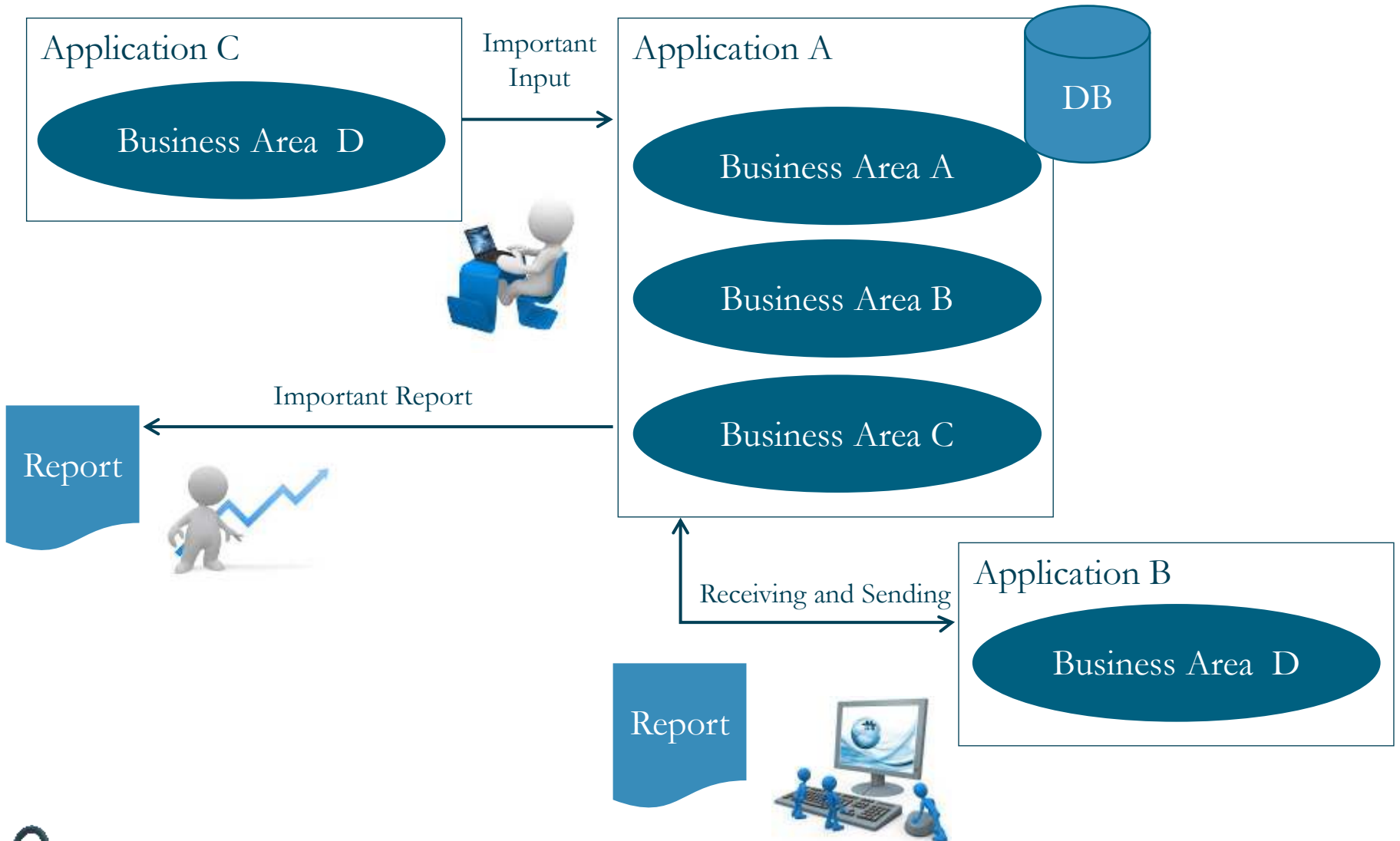
	Transactional function type		
Function	EI	EO	EQ
Alter the behavior of the application	PI	F	n/a
Maintain one or more ILFs	PI	F	n/a
Present information to a user	F	PI	PI

Identify Elementary Process

Measure Transactional Function - Activities

Compose and/or decompose into Smallest unit of activity which satisfies all of the following

Is Meaningful to the User;	Constitutes a complete Transaction;
Is Self-Contained;	Leaves the Business of the Application being Counted in a Consistent State,
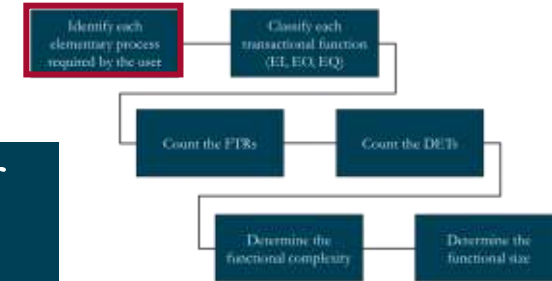
Unique Elementary Process

When compared to an Elementary Process already Identified,
Count two Similar Elementary Processes
as the same Elementary Process if they

Require the same Set of DETs
(Attributes)

Require the same Set of FTRs
(Logical Data Groups)

Require the same Set of
Processing Logic to complete
the Elementary Process

Do Not Split
an Elementary Process
with Multiple Forms of Processing Logic
into Multiple Elementary Processes

Example - Elevator

Boundary A - Elevator Interface

The perspective of the elevator from the passengers' view.

The passenger are the main user

The software that controls the elevator from an engineering perspective

Boundary B – Elevator system that makes the movements

The ability to run the elevator using AI input. Data gathering, analysis and Algorithm.
The AI Analyst is the main user

Boundary C – AI interface with AI algorithm and functionality

Not in focus

Security & Employee

UID	Description	Multiplier	Type	Complexity	DET	RET/FTE	FP	Final FP	Comment
1.1	Turn on the red light, Cap arrives, Turn off their lights, Info is saved	1	EI	HIGH	25	2	6	6	
1.X	Floor Information	1	ILF	AVG			10	10	Shared with all 1.x use cases
AI 1.X	Historical Information - maintain within AI	12.00	EI	AVG			4	48	Approximation - clarification required
AI 1.X	Historical Information's logical types	03.00	ILF	AVG			10	30	Approximation - clarification required

IP by Green

Use Case 1 - in the elevator
The 100 Feet

UC1.2 = EI-> Queue that summons request-> Determine the most appropriate cab to answer the summons-> Send a series of signals to an engine to move a cab to the floor that the summons came from when its turn comes up – uses received signals from the position marker sensor-> Send a signal to that cab's direction display to update it with the cab's current direction-> cab arrives at a floor with a pending summons - Info saved

Passenger

Elevator engineer

Boundary A
Elevator interface

UC1.1 =EI-> turn on the light ->cab arrives-> turn off their lights-> Info is saved

Elevator movement information

Floor information

Current status of CAB

Current Signal information

UC1.4 = EO-> Send a signal to the cab's door opening device to open its doors

Boundary B
– Elevator system that makes the movements

UC1.3 = EO-> Determine the most appropriate cab

Floor algorithms

Historical data

AI Analyst

Boundary C
AI interface with AI algorithm and functionality

IP by Green

Cost per Defect

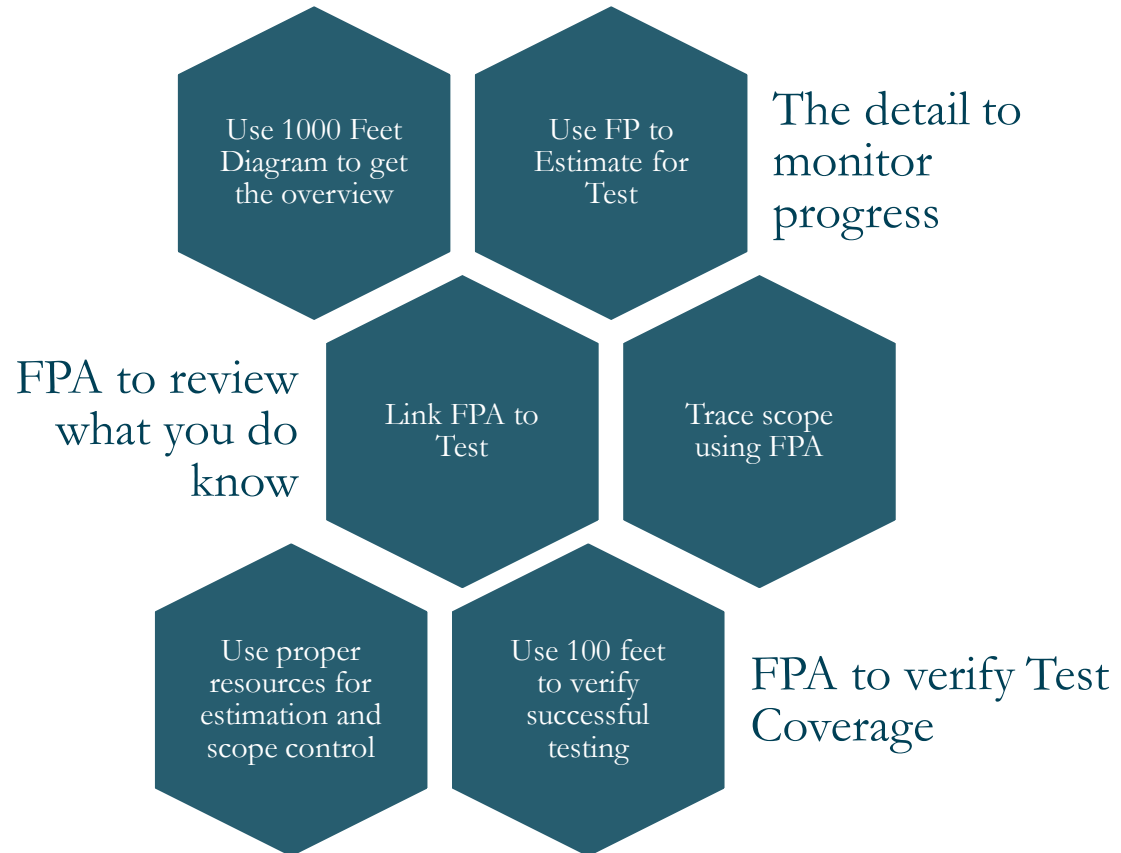| Typical data for cost per defect	
Defects Found	**Costs**
Defects Found During requirements	$250
Defects Found during design	$500
Defects Found during coding and testing	$1250
Defects Found after release	$5000

IP by Green

Defects Per FP

Software Quality for 1000 Function Points, Java and Agile Development		
Defect Potentials	**Number of Bugs**	**Defects Per FP**
Requirements defects	750	0,75
Architecture defects	150	0,15
Design defects	1000	1,00
Code defects	1350	1,35
Document defects	250	0,25
Sub Total	**3500**	**3,50**
Bad fixes	150	0,15
TOTAL	**3650**	**3,65**
Defect removal Efficiency (DRE)	**97%**	**97%**

IP by Green

Capers Jones - Software Economics and Function Point Metrics: Thirty years of IFPUG Progress Version 10.0 April 14, 2017

Acceptance Criteria and Test
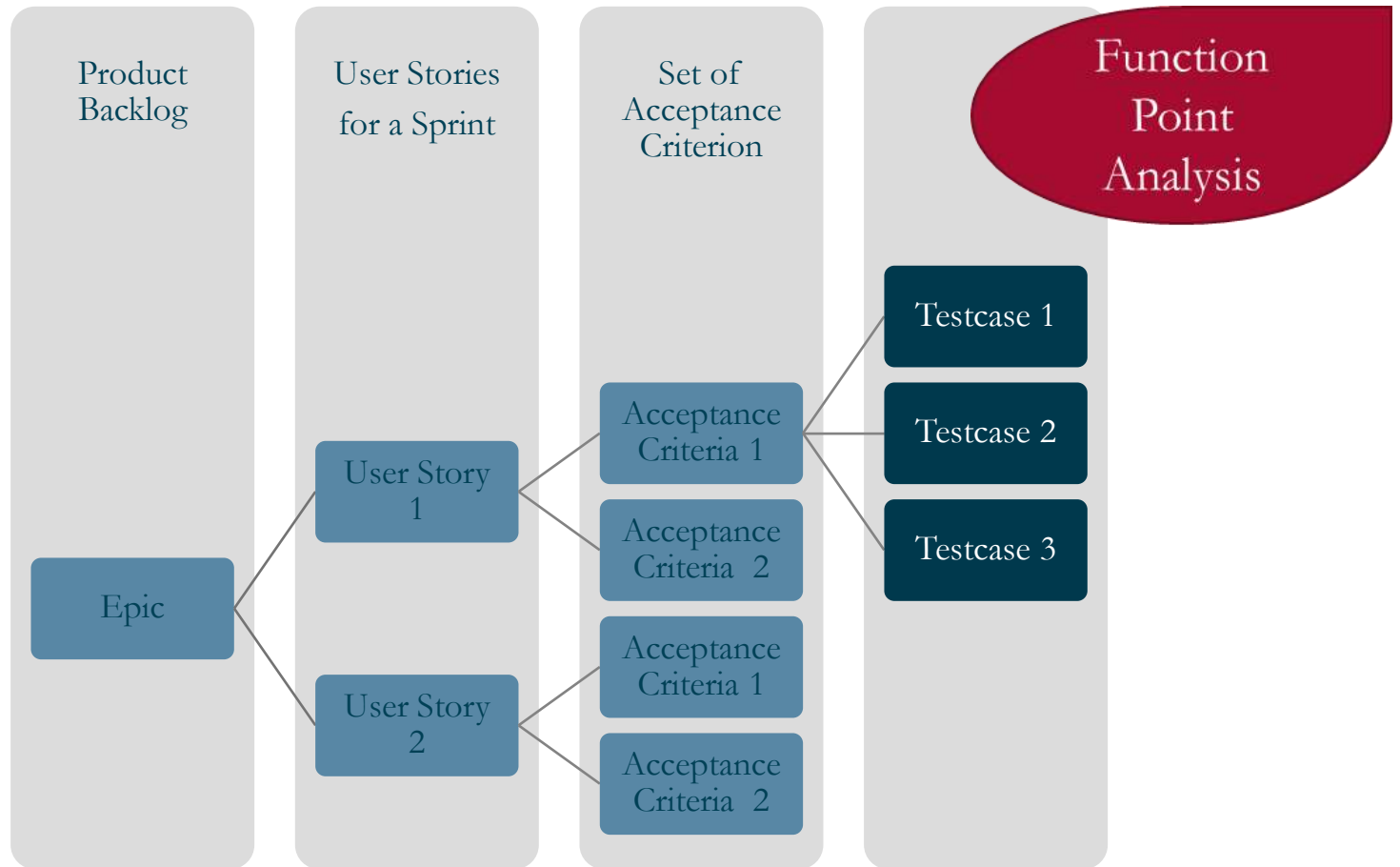
Solving the Test - tree foil knot

Use 1000 Feet Diagram to get the overview

Use FP to Estimate for Test

The detail to monitor progress

FPA to review what you do know

Link FPA to Test

Trace scope using FPA

Use proper resources for estimation and scope control

Use 100 feet to verify successful testing

FPA to verify Test Coverage

IP by Green

Final Statement: My word of wisdom

Focus on testing as a Project or path by itself

Use FPA to estimate the size and # of test cases early

Use historical data to estimate the expected amount of Defects

Use measurement to track and monitor

Use what you learned – to be better at what you do
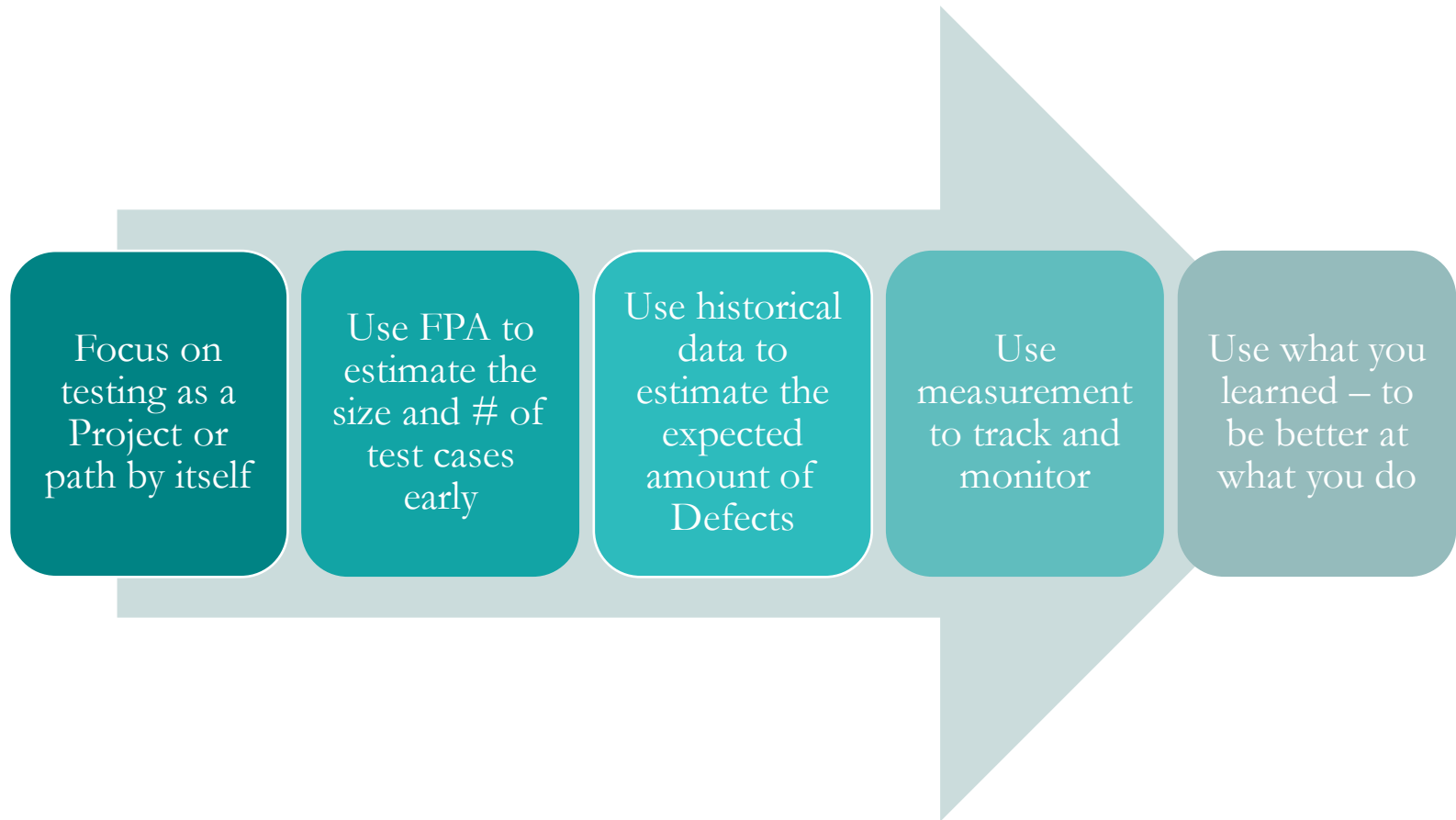
IP by Green

QUESTIONS?

Christine Green

Owner of IP$_{by}$Green

www.ipbygreen.com
Skype: christine.green
LinkedIn: christinegreendk

IP$_{by}$Green

Mobile: +45 81 72 11 22
Email: info@ipbygreen.com

What is the most influencing factor for the effort it takes to test

cheaper to fix early
knowledge
data
tools
experience
measuring
automation
executing
planning the test
business knowledge

Reach Your Potential
brightest

info@brightest.org
emilie@brightest.org
www.brightest.org

ifpug@ifpug.org
www.ifpug.org

Contact Information

BrightX Online Events

Thank you!

BrightX Online Events