



IFPUG
International **Function** Point
USERS GROUP

ISMA 2026 *Hybrid*
Powered by IFPUG

Implementing High Maturity Processes and Quantitative Management Using Function Points and CMMI

Mariano Angel Montoni

CMMI High Maturity Lead Appraiser · Promove

OCTOBER 2026



Agenda

1

High Maturity in a nutshell

What CMMI levels 4 and 5 ask for, and why FP is a natural fit

2

Understanding variation

Stable vs. capable processes, read through FP-based measures

3

Function Points across High Maturity practices

Baselines, models, planning, causal analysis and improvement – with the Radix case

4

Lessons learned and takeaways

What FP specialists should get right from day one

1. High Maturity in a nutshell





What is High Maturity?



Level 4 – Quantitatively Managed: projects are managed with statistical and other quantitative techniques to meet quality and process performance objectives.

Level 5 – Optimizing: the organization continuously improves process performance, through incremental and innovative changes, measured against business objectives.



Why Function Points are the right basic measure

Statistics need comparable data.

Raw effort, defects and cost depend on how big the product is. Dividing them by functional size makes projects comparable.

Function Points are:

- Independent of technology and language
- Based on a public standard (IFPUG / ISO)
- Known early, from requirements
- Understood by business and IT

h / FP

Productivity

Effort per function point

$defects / FP$

Defect density

Quality delivered to the customer

$cost / FP$

Unit cost

Link to price and margin

$criteria / FP$

Test depth

Test criteria per function point



Managing projects: Level 3 vs. Levels 4 and 5

Up to Level 3

- › Plan with estimates (size, effort, cost)
- › Track actuals against the plan
- › React when a deviation appears
- › Lessons learned are mostly qualitative

Levels 4 and 5

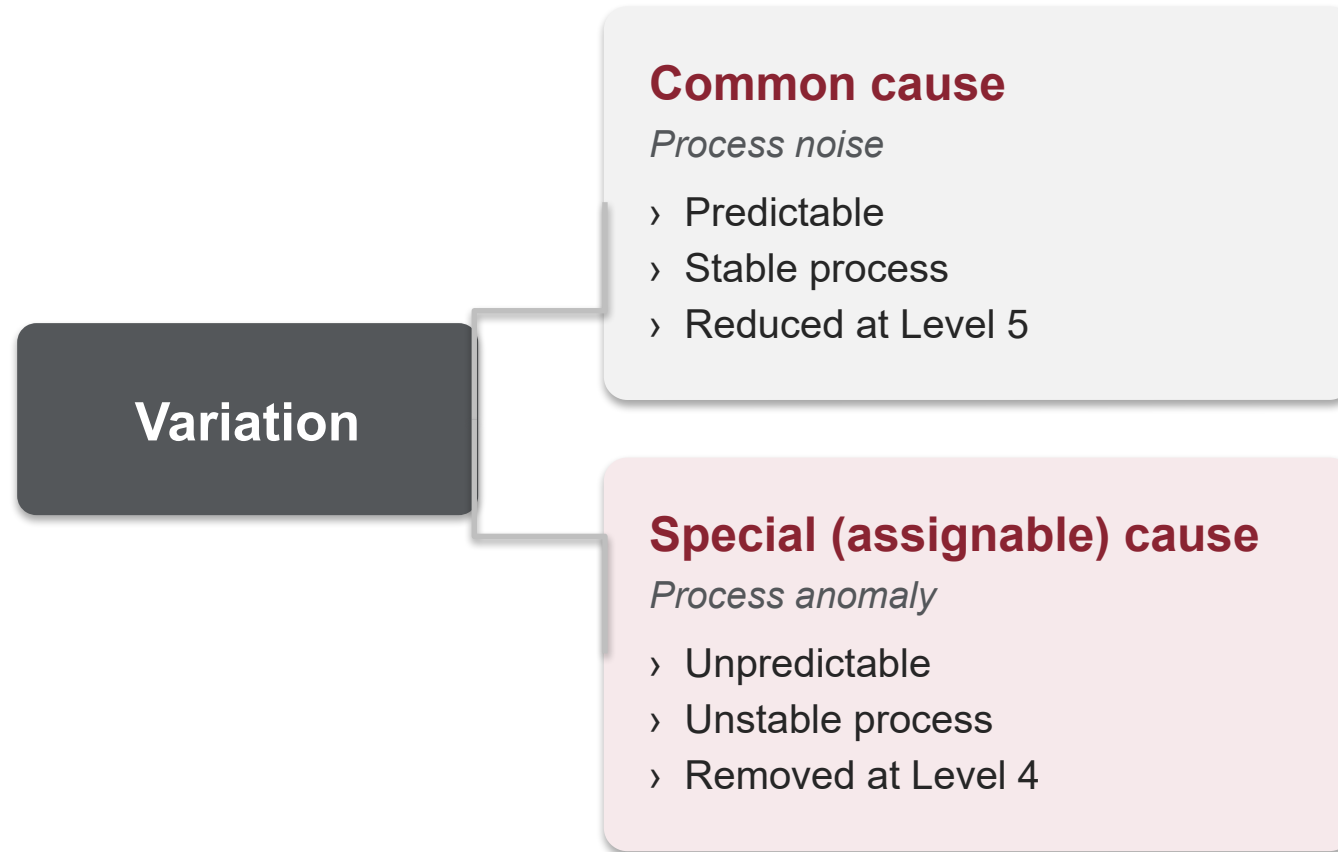
- › Set objectives from business goals, in FP terms
- › Use baselines to know what is normal
- › Use models to predict if objectives will be met
- › Act early; confirm improvements with statistics

2. Understanding variation





Every process varies: know which variation you have



Real case: Radix

The process performance baselines showed high variation in productivity:

- Requirements Analysis: 0.46 to 5.09 h/FP
- Testing: 0.80 to 3.55 h/FP

Wide limits mean effort is hard to predict and plans are risky.

Ways to improve: centre the process and reduce its spread.

Source: CMMI Institute – Radix case study



Is the process predictable? Radix baselines in h/FP

Requirements Analysis (S1)



Testing (S4)



0 1 2 3 4 5 6
Productivity (h/FP) · bar = LCL to UCL · ◆ = mean

Voice of the process

Control limits from I-MR charts of productivity (h/FP) per subprocess.

Voice of the customer

Realistic commitments on effort and schedule.

Stable vs. capable

Stable = only common causes. Capable = stable and meets the objective. Radix's limits were too wide to plan with confidence.

For a 1,000 FP project, Requirements Analysis alone could take 460 to 5,100 hours.

Same process, same counting rules: the spread is the problem, not the size.

Source: CMMI Institute – Radix case study (baselines before the improvement)

3. Function Points across High Maturity practices





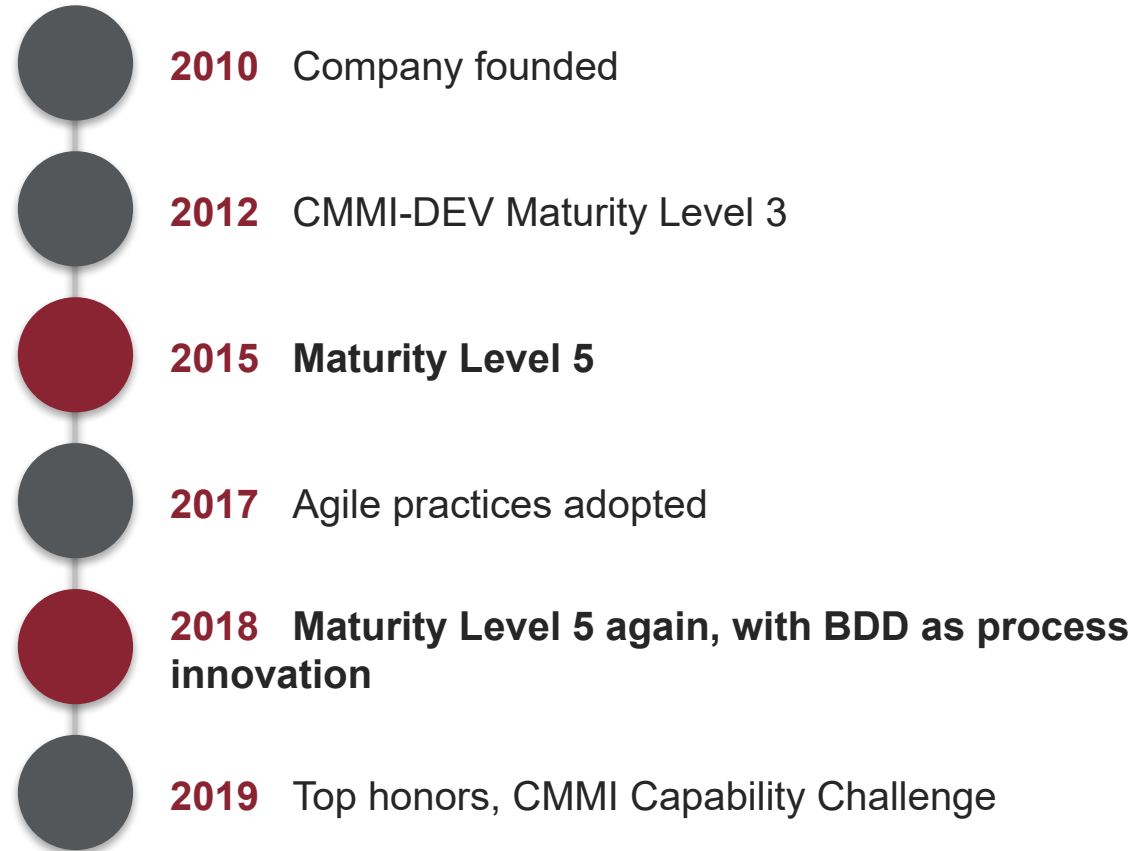
Real case: Radix

Who

Brazilian engineering and software company, founded in 2010, serving primary process industries in Brazil and abroad.

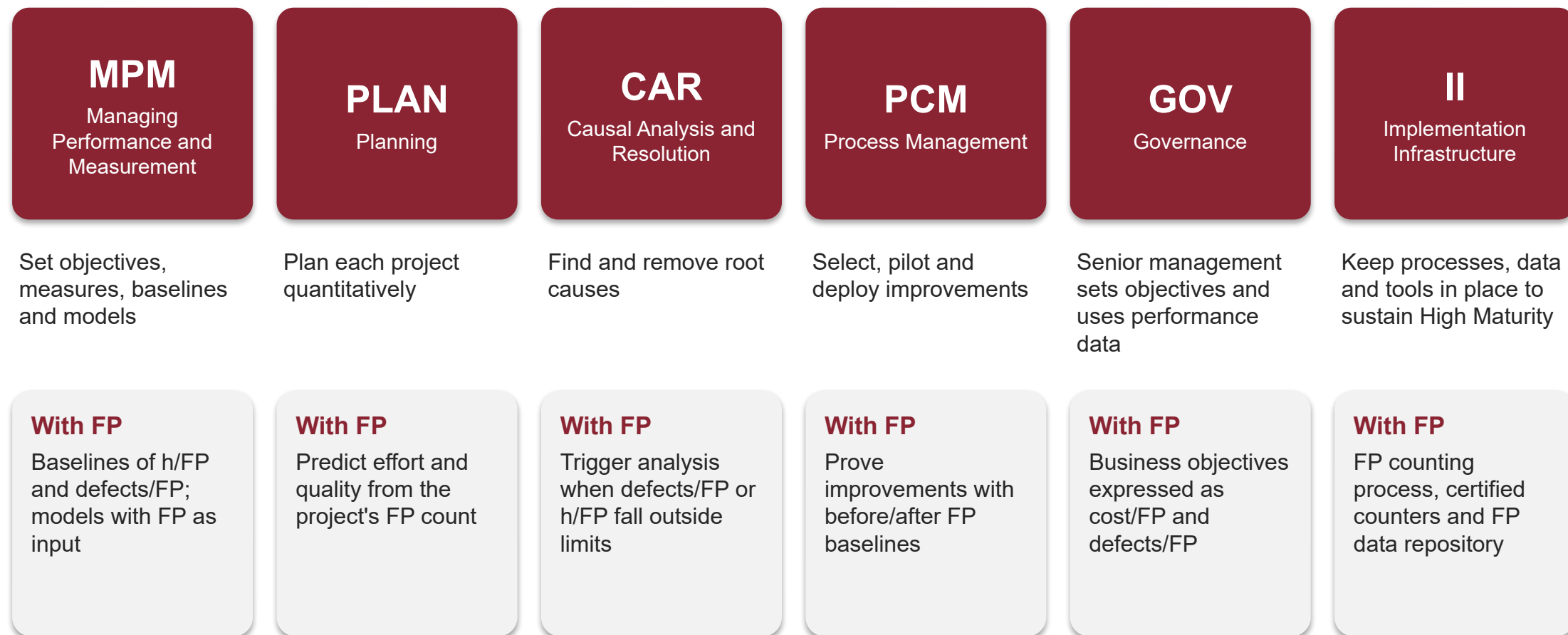
Business needs

- Be a benchmark partner for clients' critical technology issues
- Improve team communication and productivity
- Gain credibility in the U.S. market through product quality and project performance



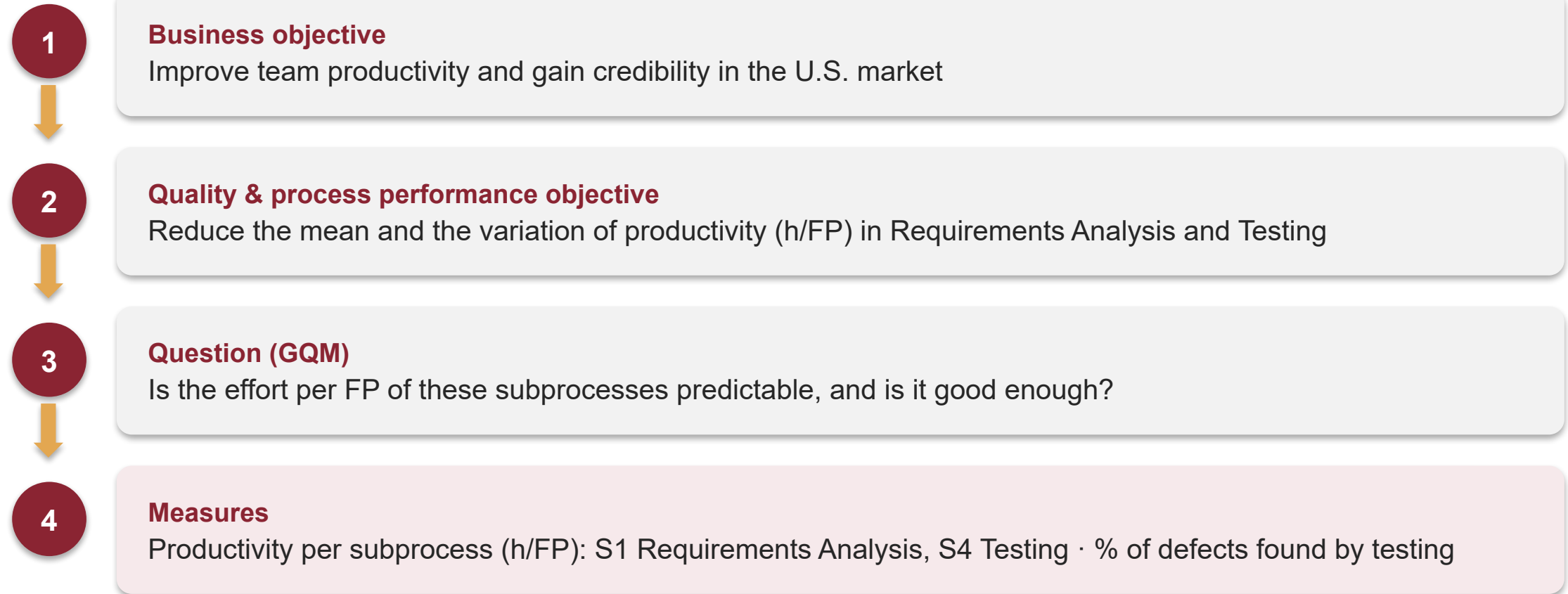
Source: CMMI Institute – Radix case study

Where Function Points fit in CMMI High Maturity





MPM: from business objectives to FP-based measures



Based on the Radix case (CMMI Institute); objective wording simplified

MPM: process performance baselines in FP

Measure	Formula	Example use
Productivity	Effort hours ÷ FP	Estimate effort; detect unusual projects
Delivery rate	FP ÷ calendar month	Predict schedule and time-to-market
Delivered defect density	Production defects ÷ FP	Quality objective for the customer
Defect removal efficiency	Defects before release ÷ total defects	Effectiveness of reviews and tests
Test depth	Test criteria ÷ FP	Input to quality prediction models
Unit cost	Project cost ÷ FP	Link process performance to price and margin

Good baselines

- Come from stable processes (or state the uncertainty)
- Use the same counting rules in every project
- Split by natural strata: development vs. enhancement, platform, domain
- Are refreshed often as new projects close
- Use hypothesis tests to compare groups (t-test, Mann-Whitney, ANOVA)

MPM: a process performance model using FP

$$\text{Acceptance test pass rate} = 0.821 + 0.0682 \times (\text{test criteria per FP})$$

75%

of the variation explained (R^2)

0.001

p-value: the model is significant

+6.8 p.p.

percentage points of pass rate for each extra test criterion per FP

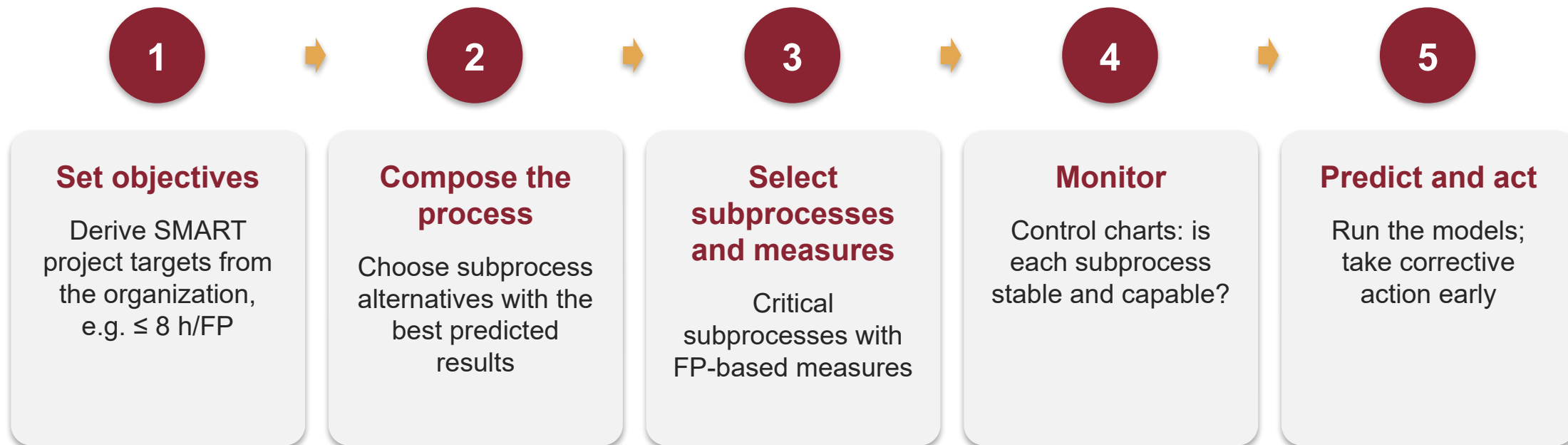
Y (outcome): acceptance test pass rate — known only at the end.

X (factor): test criteria per FP — controlled during the project.

Other FP-based factors to model: requirements volatility (% FP changed), staff turnover, code complexity, review coverage.

Source: Promove High Maturity training material (simple linear regression, 11 projects)

PLAN: managing a project quantitatively with FP

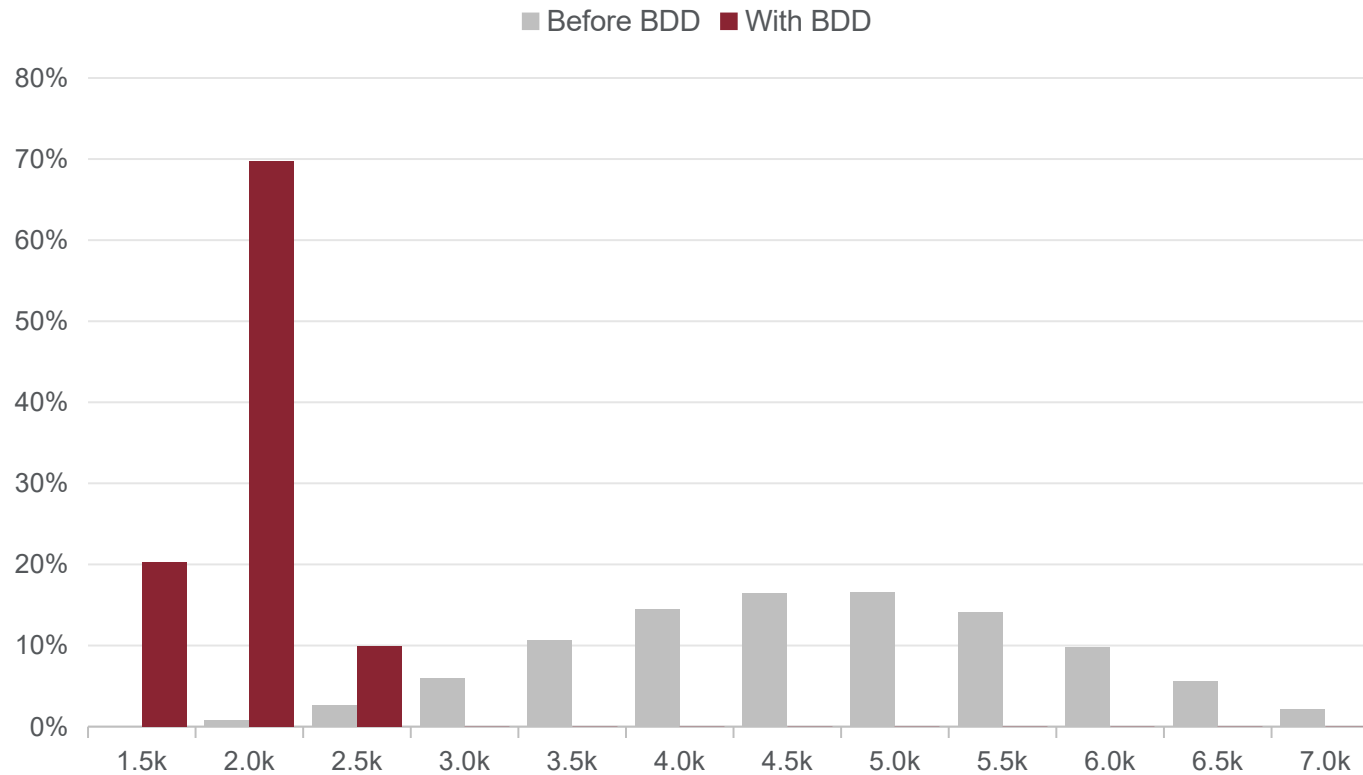


Tip for small projects: aggregate data from similar, homogeneous projects (same domain, platform and team profile).



PLAN: what is the chance of meeting the target?

Simulated effort, Requirements Analysis + Testing, 1,000 FP (% of runs)



Monte Carlo (20,000 runs) on Radix baselines; project size and budget are illustrative

Inputs (Radix baselines)

Size: 1,000 FP · Budget: 5,000 h · Min / likely / max = LCL / mean / UCL in h/FP

51%

chance to meet the budget with the old process

>99%

with the BDD subprocess



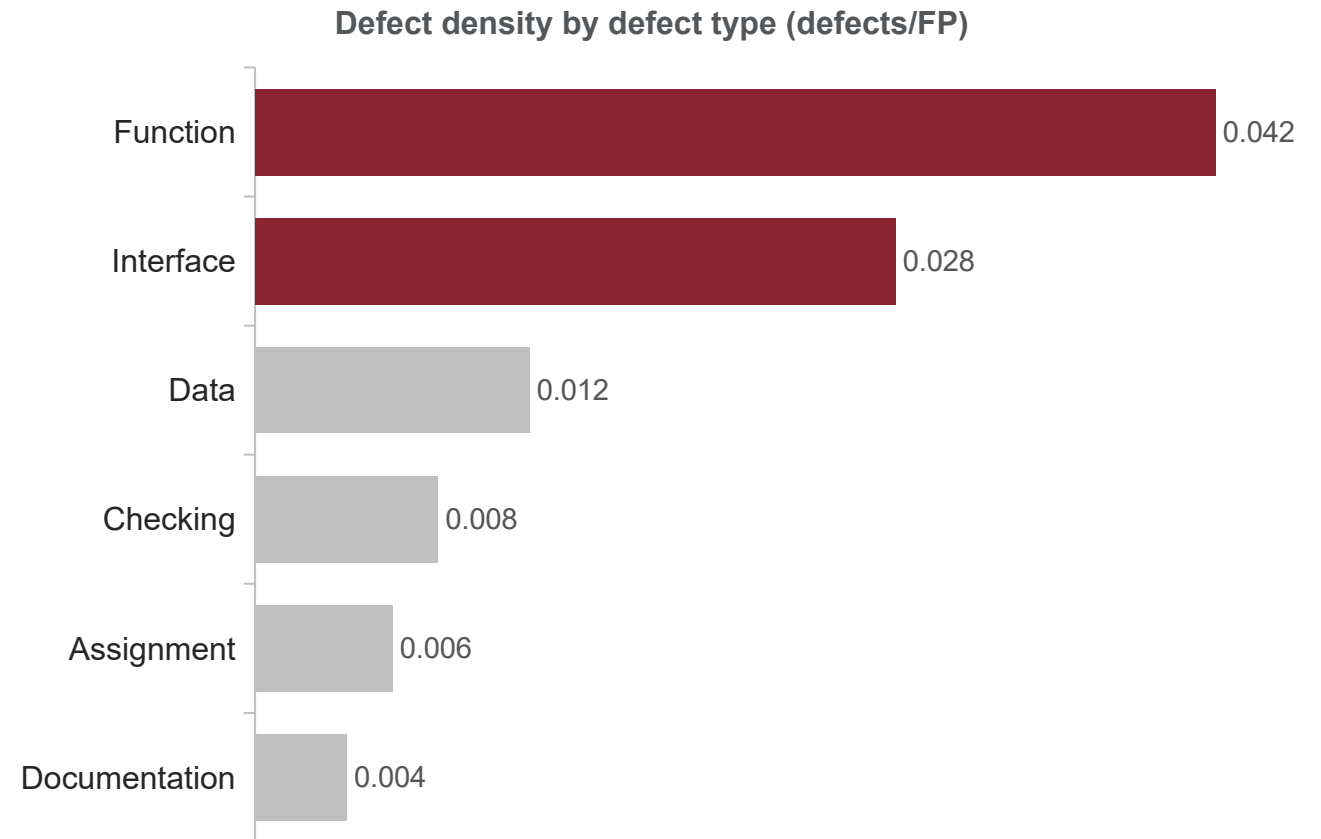
CAR: finding root causes with FP-normalized data

When to start a causal analysis

- A point outside the control limits (h/FP, defects/FP)
- A model predicts an objective will be missed
- More defects per FP escape than expected
- Performance is much better than expected

How

- Pareto and control charts to select the outcome
- Ishikawa or 5 Whys to find the root cause
- Record actions, cost and effect for the whole organization



Illustrative data – orthogonal defect classification



PCM: measuring the improvement – Radix and BDD

S1 Requirements – before



S1 Requirements – BDD



S4 Testing – before



S4 Testing – BDD



0 1 2 3 4 5 6

Productivity (h/FP) · bar = LCL to UCL · ◆ = mean

To prove it: compare before/after baselines with hypothesis tests (e.g. t-test or Mann-Whitney; $p < 0.05$), then update baselines and models.

Source: CMMI Institute – Radix case study

–37.8%

Requirements Analysis effort per FP
2.772 → 1.724 h/FP

–81.9%

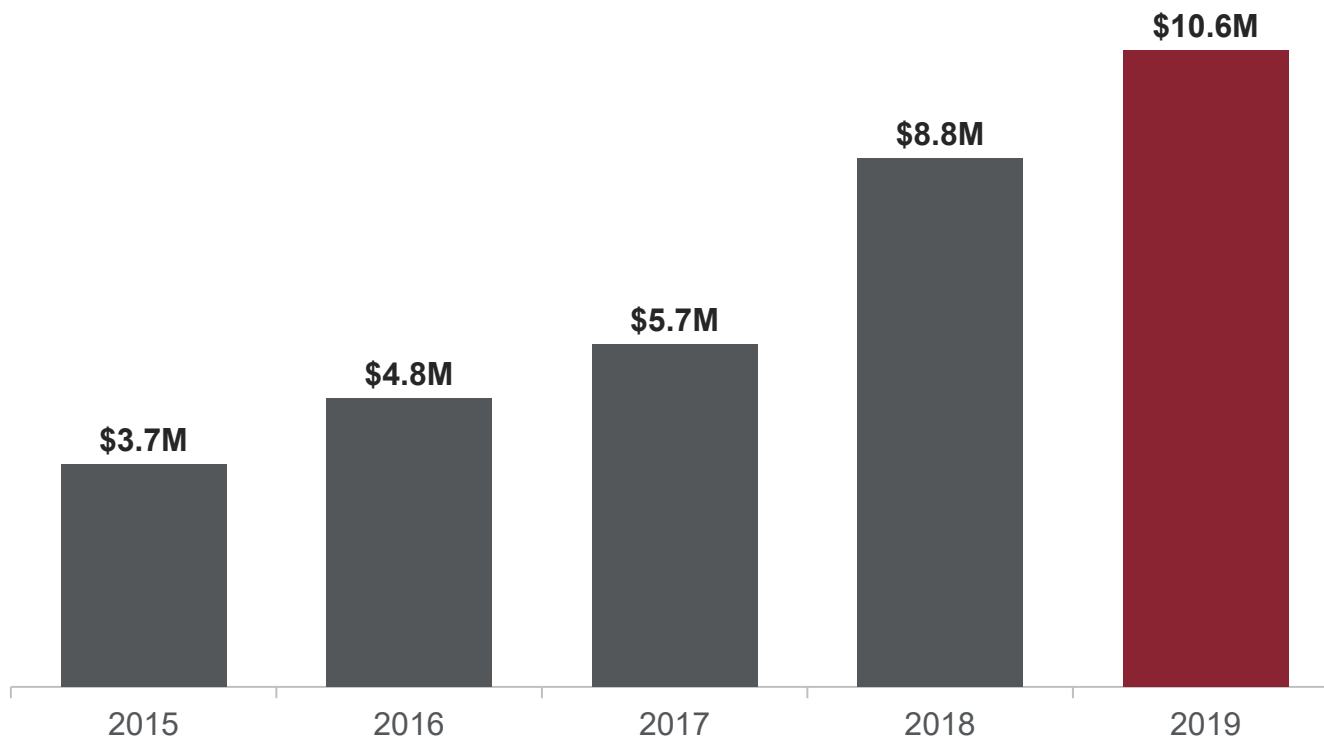
Testing effort per FP
2.173 → 0.392 h/FP

90%

of all defects found by testing

GOV, MPM and PCM: linking results to business strategy

Radix revenue (US\$ million)



Source: CMMI Institute – Radix case study · values converted at R\$3.00 per US\$

30%

ROI 2015–2016, after High Maturity processes

55%

ROI 2017–2018, after adding agile practices

-40%

cost of the High Maturity initiative vs. the earlier Level 3 initiative

< 2 years

to a positive return on the investment

4. Lessons learned and takeaways





Lessons learned for FP specialists

1

Count consistently

Same IFPUG rules, guidelines and certified counters in every project.

2

Know which count you use

Estimated (early) vs. final count; record both and when each was taken.

3

Keep data homogeneous

Separate development and enhancement; group by platform and domain.

4

Don't over-aggregate

Mixing contexts hides problems; aggregate only similar projects.

5

Collect enough data

Too few points make baselines weak and tests inconclusive.

6

Check the counts first

An out-of-limit point is often a counting error, not a process problem.



Key takeaways

FP makes data comparable

Normalizing effort, defects and cost by FP is the base for baselines and SPC.

Predict, don't just track

FP-based models tell a project early if it will meet its objectives.

Prove improvements

Before/after baselines and hypothesis tests separate real change from noise.

Speak the business language

Cost per FP and defects per FP connect process performance to revenue and cost.



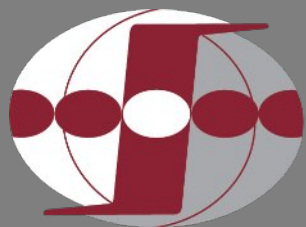
Thank you! Questions?

Mariano Angel Montoni

CMMI High Maturity Lead Appraiser · Co-founder, Promove

www.promovesolucoes.com

contato@promovesolucoes.com



IFPUG

International Function Point Users Group

Much more at
<http://www.ifpug.org>

Email: ifpug@ifpug.org

Let's connect!

