



EXPERIENCE OF USING COSMIC SIZING IN AGILE PROJECTS

Alain Abran, Onir Demirors, Charles Symons

November 2017

Background

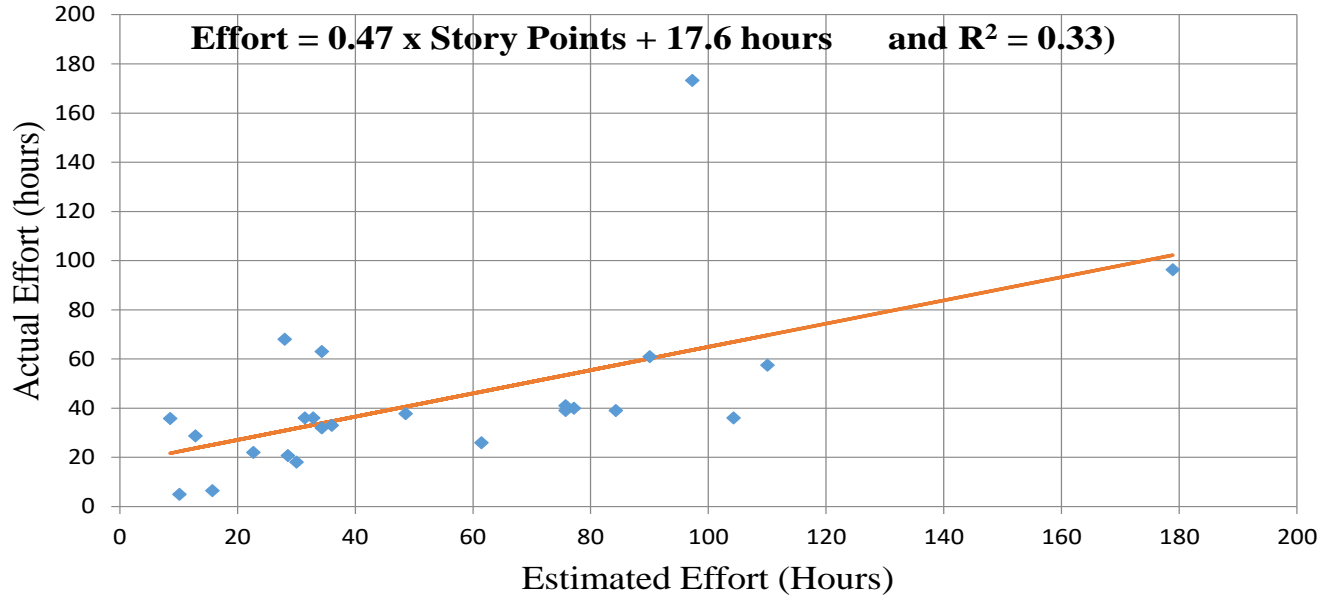
- Organizations experienced in Agile methods are starting to realise the limitations of ‘Story Points’ (SP)
 - The process of planning a sprint via Story Points (e.g. ‘Planning Poker’) may be good, but resulting ‘velocity’ measures only have any meaning for individual Teams
 - so it’s not possible to compare performance and track progress across Teams.
 - SP-based measurements are poor for early effort estimation, and no help for organizational learning
- Studies are now showing the benefits of using ‘COSMIC Function Points’ (an objective measure of software size - see last slide) instead of subjective Story Points
- This paper summarises the findings of four studies showing that Agile sprint/iteration sizes measured using CFP correlate much better with effort than do SP sizes

Case: Canadian supplier of security and surveillance software systems

- A customer request for new or changed function is called a ‘task’
- Scrum method used; iterations last 3 – 6 weeks
- Teams estimate tasks within each iteration in SP, and convert directly to effort in work-hours
- 24 tasks in nine iterations were analysed *
 - Estimated task sizes in SP, converted to estimated effort
 - Actual effort also recorded
 - Each task also measured in CFP

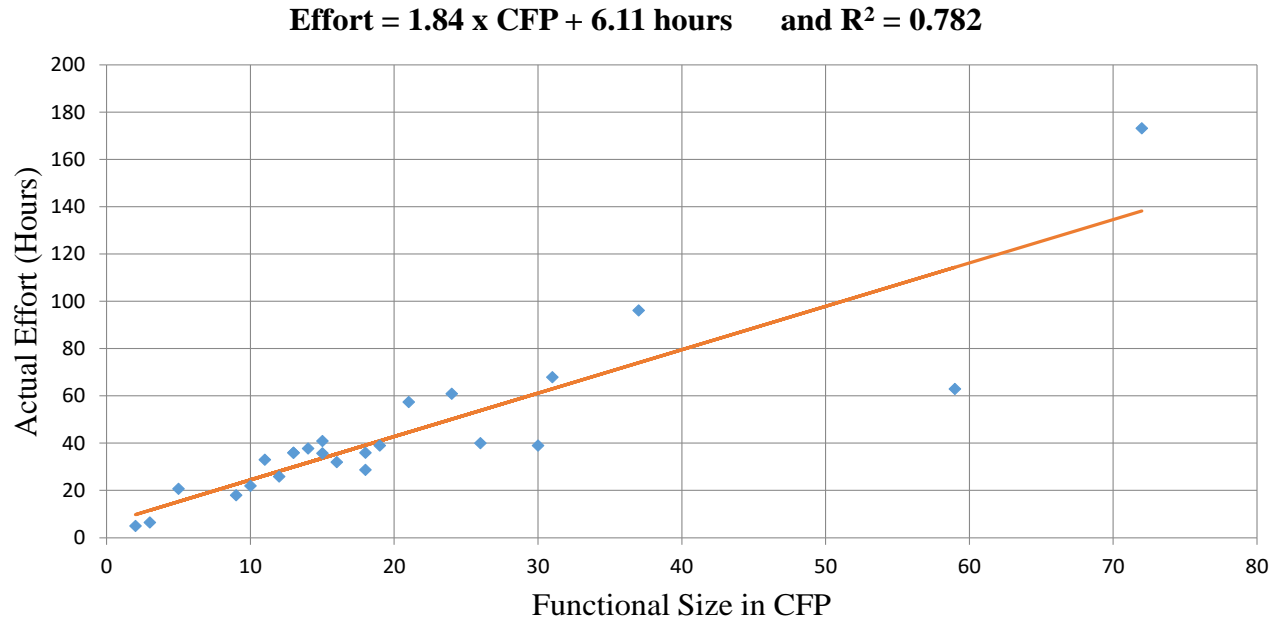
* ‘Effort Estimation with Story Points and COSMIC Function Points - An Industry Case Study’, *Christophe Commeyne, Alain Abran, Rachida Djouab*. ‘Software Measurement News’. Vol 21, No. 1, 2016. <https://cosmic-sizing.org/publications/effort-estimation-story-points-cosmic-function-points-industry-case-study/>

A best-fit straight line would be a poor predictor of effort from SP sizes



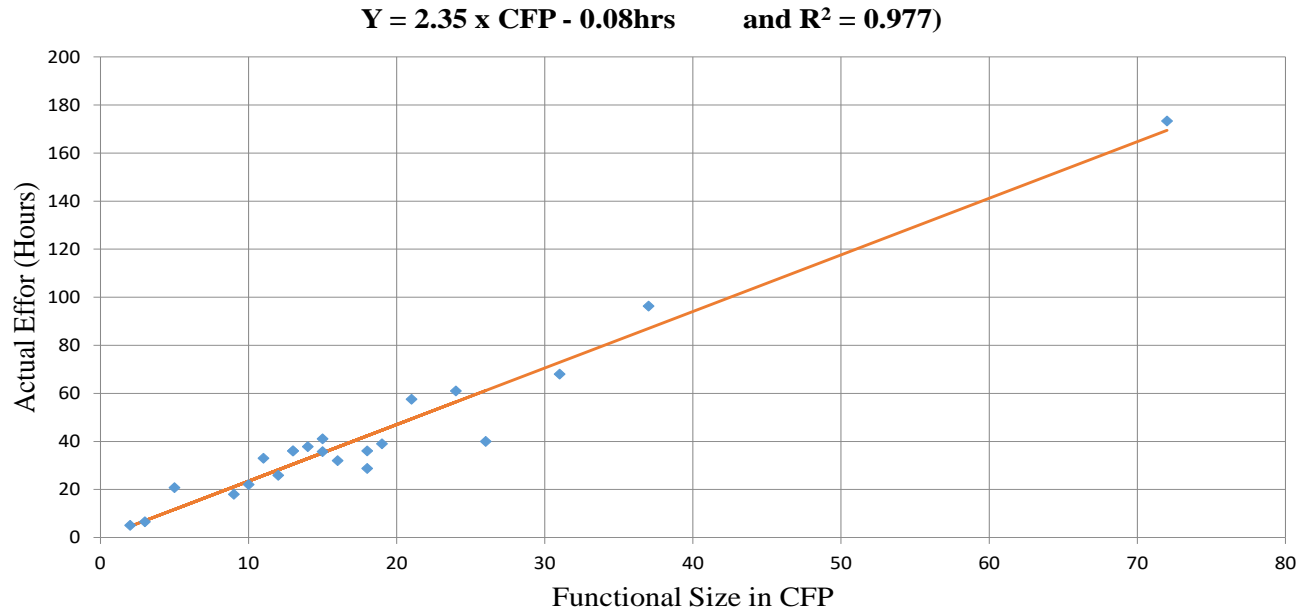
Notice the wide spread and the 17.6 hours 'overhead' for zero CFP

The Effort vs CFP size graph (24 tasks) shows a good fit, but two outliers remain



Two tasks with low effort/CFP were found to involve significant software re-use, so were rejected as outliers

Now we have a good effort vs CFP correlation (22 tasks), usable for predicting task effort



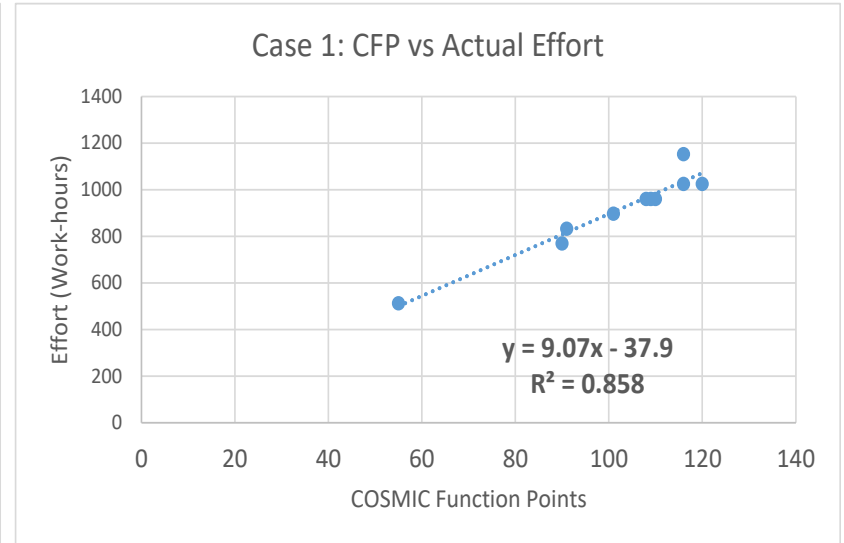
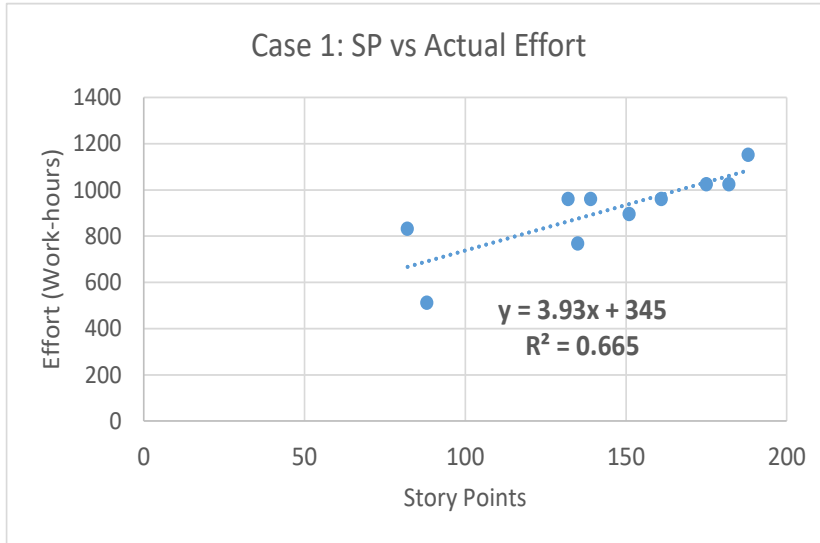
R-squared = 0.977, intercept $\approx (0,0)$

Large Turkish supplier of security software*

- SCRUM method used
- Web portal project for one Team (6 developers, 2 testers)
- Ten 3-week Sprints analysed
- A planning meeting is held for each Sprint when sizes are estimated in SP and Stories are allocated to Sprints
- CFP sizes were measured retrospectively from ‘mature’ documentation in JIRA
- (CFP measurement effort averaged 4.1 hours/Sprint, = 25 CFP/hour)

* ‘Effort estimation for Agile software development. Comparative case studies using COSMIC Function Points and Story Points’. Murat Salmanoglu, Tuna Hacaloglu, Onur Demirors. Ankara, Turkey. IWSM/Mensura Conference, Gothenburg 2017, <https://doi.org/10.1145/3143434.3143450>

Completed CFP correlate much better with Actual Effort than do Story Points

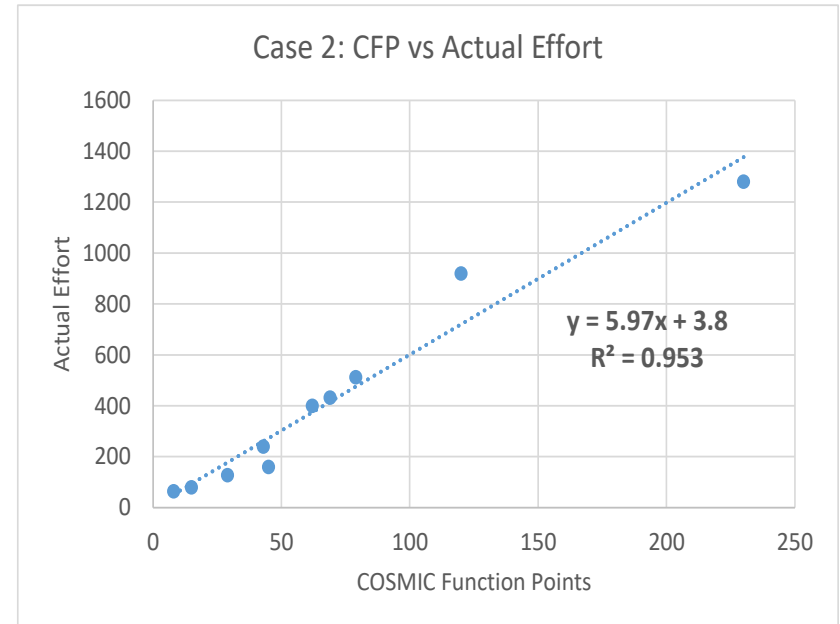
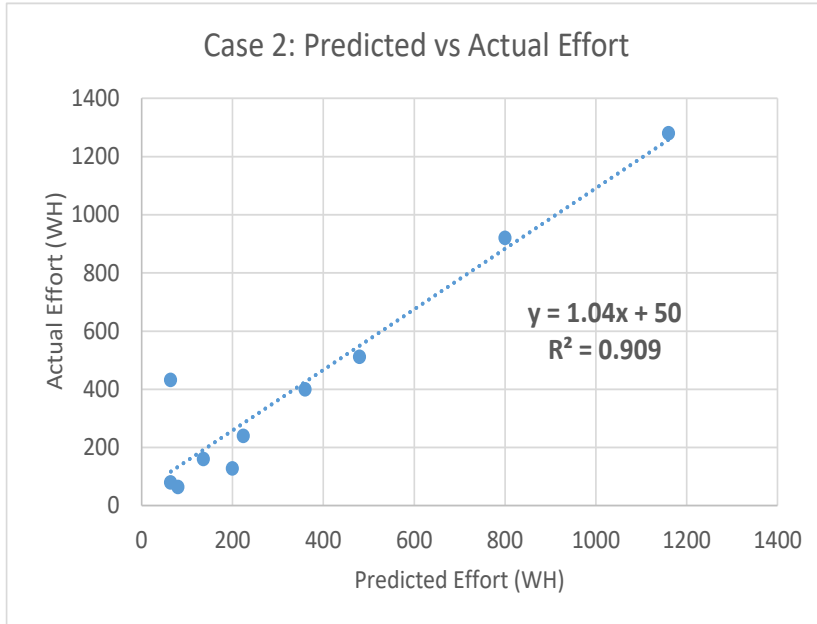


CFP vs Actual effort has much better R^2 and a much smaller intercept for CFP = 0

Large Turkish software organization mainly supplying the telecoms industry*

- 500 Developers using Agile methods
- Ten Change Request 'Projects' were studied for one specific development team
- Story Points are estimated by experts and converted directly to 'Predicted effort'
- CFP sizes measured retrospectively from the same 'not mature' CR documents + other information
- (Measurement effort averaged 1 day/project, ~9 CFP/WH)

Completed CFP correlate better with Actual Effort than does Predicted Effort (\equiv SP)

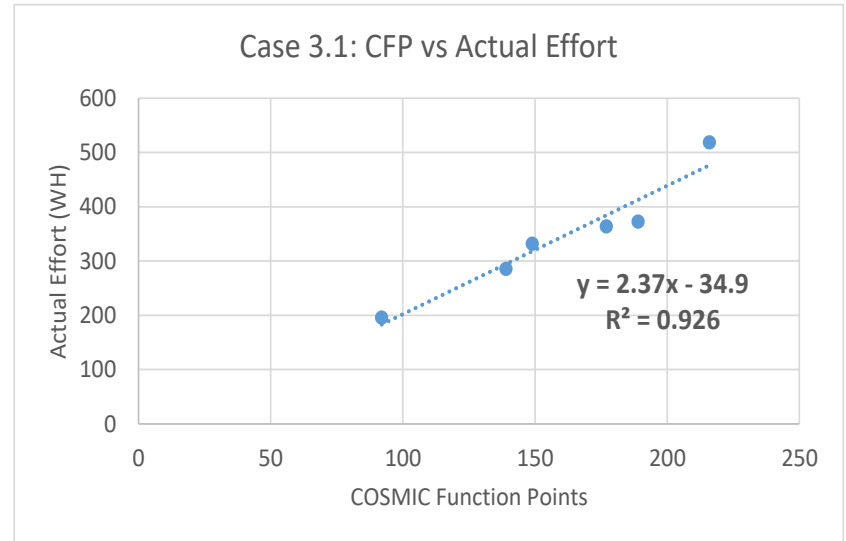
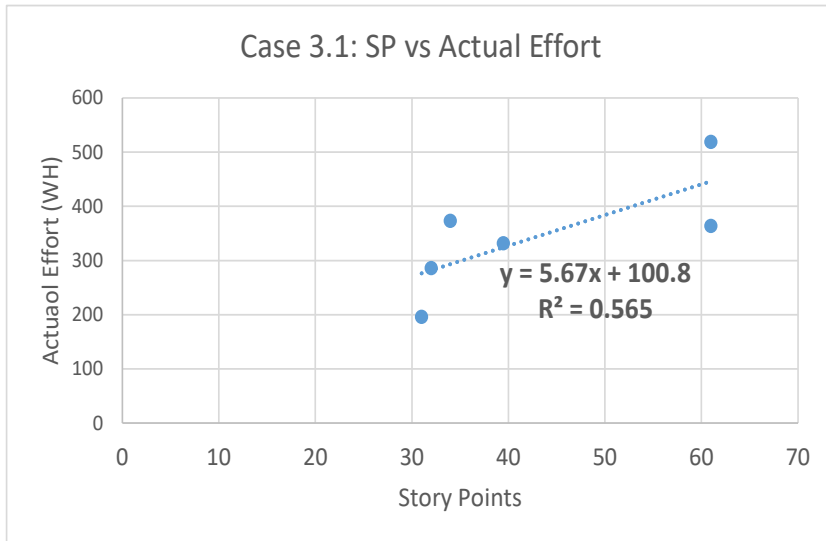


CFP vs Actual effort has better R^2 and much lower intercept for CFP = 0

Large Turkish software developer, supplying mainly to finance and banking industry*

- SCRUM method used
- Requirements documentation ‘lacking’
- Story Points are directly converted to estimated effort, but the latter data were not available
- CFP sizes were measured retrospectively
- Results shown here are for 6 projects that used the same C# technology

Completed CFP correlate much better with Actual Effort than do Story Points



CFP vs Actual effort has much better R^2 and much better intercept for CFP = 0

Finally, a User view of ‘COSMIC for Agile’

- *“We have found that adopting this approach provides us with excellent predictability and comparability across projects, teams, time and technologies.”*
- *The reality of achieving predictable project performance has driven me to investigate many methods of prediction. COSMIC is the method that lets me sleep at night.”*

Denis Krizanovic, Aon Australia, August 2014

Conclusion: CFP sizes correlate very well with effort - much better than Story Points

- Correlations of post-calculated CFP sizes with actual effort versus SP/effort correlations:
 - higher R-squared (better)
 - intercepts for zero CFP much closer to zero effort (more realistic)
- See the original papers for other interesting results

Conclusion: CFP can beneficially replace SP, with no other changes to Agile practices

Story Points

- In practice, a subjective estimate of relative effort
- Have any meaning only within a project team
- Poor for estimating total project effort
- No guidance on how to deal with Non-Functional Requirements.

COSMIC Function Points

- An objective, ISO Standard measure of functional size
- Sizes meaningful across projects and teams
- Good for estimating at all levels (US, Sprint, Release, System)
- Method advises how to deal with NFR

Getting Agile teams to accept measurement is the biggest challenge

Footnote: be careful about comparing the productivity figures of the four datasets

‘Product Delivery Rate’ values (i.e. the slopes of the fitted CFP vs effort lines) for the four datasets vary from 2.35 to 9.1 work-hours/CFP.

The following factors are known to influence these ‘PDR’ figures:

- different levels of decomposition of the software
- different activities included in the effort figures
- different work mixes (new requirements, change requests)
- varying requirements documentation quality (but no measures of product quality)
- varying amounts of functional or code re-use
- different application domains, technologies, work practices
(and maybe) different skill-levels, hence different real productivities

For more on the COSMIC method of measuring a size of software requirements:

Download (free!)

- ‘Introduction to the COSMIC method of measuring software’ <https://cosmic-sizing.org/publications/introduction-to-the-cosmic-method-of-measuring-software-2/>
- ‘The ‘COSMIC Functional Size Measurement Method, version 4.0.2: Measurement Manual’ <https://cosmic-sizing.org/publications/measurement-manual-v4-0-2/>
- ‘Guideline for the use of COSMIC FSM to manage Agile projects’: <https://cosmic-sizing.org/publications/guideline-for-sizing-agile-projects-with-cosmic/>