SOFTWARE PROJECTS ESTIMATION & CONTROL: VERSATILITY & CONTRIBUTIONS OF COSMIC FUNCTION POINTS

Alain Abran
with C. Symons, C.Ebert, F.Vogelezang, H.Soubra

ICEAA 2017 Professional Development & Training Workshop
Portland, Oregon (USA), June 6-9, 2017
Presenter background - Alain Abran

20 years

- Development
- Maintenance
- Process Improvement

20 years

+ 40 PhD

ISO: 19761, 9126, 25000, 15939, 14143, 19759

© Copyrights 2017: Alain Abran
Agenda

1. Software effort estimation & software size
2. COSMIC: 2nd generation of Function Points
3. Versatility of COSMIC Function Points
4. Contributions of COSMIC to Estimation models
5. Early & Quick COSMIC sizing at estimation time
6. Automation of COSMIC Function Points
7. Summary

© Copyrights 2017: COSMIC and authors

ICEAA Portland - Oregon (USA), June 2017
The Cone of Uncertainty across the Project Lifecycle

Range of expected variations in ‘estimation’ models across the project life cycle
Adapted from Boehm (2000), Fig. 1.2

© Copyrights 2017: COSMIC and authors
Productivity Models: Built with Data from Completed Objects

You build estimation models with completed projects (with almost no uncertainty in the inputs)

© Copyrights 2017: COSMIC and authors
Estimation Foundations: Productivity Models with Uncertainties in the Inputs

- You do estimation upfront with a lot of uncertainty
- What do you have upfront as available for estimation purposes?
Software: What is Available & Measurable Across the Lifecycle?

Margin of error = orders of magnitude!
### Software Sizing Options across the Lifecycle?

<table>
<thead>
<tr>
<th>Sizing method options:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Lines of code:</strong></td>
<td>X Can’t estimate until software designed</td>
</tr>
<tr>
<td></td>
<td>X Technology-dependent, no standards</td>
</tr>
<tr>
<td><strong>Usecase Points, Object Points, ..</strong></td>
<td>X Technology dependent, no standards,</td>
</tr>
<tr>
<td></td>
<td>X Mathematical validity?</td>
</tr>
<tr>
<td><strong>Story Points</strong></td>
<td>X Entirely subjective</td>
</tr>
<tr>
<td><strong>Functional size</strong></td>
<td>✓ International standard methods</td>
</tr>
<tr>
<td>✓ (Function Points):</td>
<td>✓ Technology-independent</td>
</tr>
</tbody>
</table>

© Copyrights 2017: COSMIC and authors
Measure & track software requirements

Estimating budgeting, etc.

Project productivity & performance

Organization Data Repository

Variability (%)

Marginal of effort = ratio of implication

CH01FG05-50

0

1

2

4

-25

-50

0

1

2

5

3

1

0

Time

Original Estimate

clipse software requirements: the Foundations for Estimation & Benchmarking Studies

© Copyrights 2017: COSMIC and authors

ICEAA Portland - Oregon (USA), June 2017
Agenda

1. Software Effort Estimation & Software Size
2. COSMIC: 2nd generation of Function Points
3. Versatility of COSMIC Function Points
4. Contributions of COSMIC to Estimation models
5. Early & Quick COSMIC sizing at estimation time
6. Automation of COSMIC Function Points
7. Summary
1st Generation of Function Points: Weights

Inputs - Matrix

Output & Enquiries - Shared Matrix

Transactions: weights in FP (Function Points)
First Generation of Function Points: Step Functions

Function Points (FP)

Key limitations:
- Only 3 values
- Limited ranges (min, max)
- No single measurement unit of 1 FP!

3-step size range for the IFPUG External Input Transactions

© Copyrights 2017: COSMIC and authors

ICEAA Portland - Oregon (USA), June 2017
1st Generation of Function Points

Function Points (FP)

3 FP

4 FP

6 FP

= ?
1st & 2nd generation of Function Points

1st generation

- Allan Albrecht FPA
- Feature Points
- 3-D FP's
- MkII FPA

2nd generation

- ISO ‘FSM’ Standard 14143
- COSMIC FFP v. 2.0
- COSMIC v. 4.0.1
- Full FP’s v.1
- IFPUG 4.1
- IFPUG 4.3
- MkII FPA v.1.3
- IFPUG 4.0
- 1st & 2nd generation of Function Points

© Copyrights 2017: COSMIC and authors

ICEAA Portland - Oregon (USA), June 2017
Every software is different, but what is common:

- In all software?
  - In different types of software?
  - In very small or extremely large software?
2nd Generation of Function Points: All software does this!

Functional Users:
- Humans
- Hardware devices
- Other software

The ‘Data Movement of 1 data group’ is the unit of measurement: 1 CFP (1 CFP = 1 COSMIC Function Point)

COSMIC view of software

© Copyrights 2017: COSMIC and authors
All Software also does this…

Event → Functional User → Functional Processes → Data Movements

© Copyrights 2017: COSMIC and authors

Presented at the 2017 ICEAA Professional Development & Training Workshop
www.iceaaonline.com/portland2017
2nd Generation with COSMIC

COSMIC Function Points (CFP)

No arbitrary max

Largest observed functional processes:
- In avionics > 100 CFP
- In banking > 70 CFP

A single CFP exists & is well defined
Example 1: Intruder Alarm System - Requirements

The embedded alarm software

Input devices (functional users)
- Keypad
- Power voltage detector
- Front door sensor
- Movement detectors

Software Boundary

Output devices (functional users)
- External alarm
- Internal alarm
- 2 x LED’s

Persistent storage
Example 1: Intruder Alarm System - COSMIC size

Functional process: Possible intruder detected.
Triggering event: Door opens whilst alarm system is activated.

<table>
<thead>
<tr>
<th>Data Movement</th>
<th>Functional User</th>
<th>Data Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry</td>
<td>Front-door sensor</td>
<td>‘Door open’ message (triggering Entry)</td>
</tr>
<tr>
<td>Read</td>
<td>- / Occupant</td>
<td>PIN (from persistent storage)</td>
</tr>
<tr>
<td>Exit</td>
<td>Green LED</td>
<td>Switch ‘off’ command</td>
</tr>
<tr>
<td>Exit</td>
<td>Red LED</td>
<td>Switch ‘on’ command</td>
</tr>
<tr>
<td>Exit</td>
<td>Internal siren</td>
<td>Start noise command</td>
</tr>
<tr>
<td>Entry</td>
<td>Keypad</td>
<td>PIN (If the wrong code is entered, the user may enter the PIN two more times but the process is always the same so it is only measured once.)</td>
</tr>
<tr>
<td>*</td>
<td>Green LED</td>
<td>Switch ‘on’ command (after successful entry of PIN)</td>
</tr>
<tr>
<td>*</td>
<td>Red LED</td>
<td>Switch ‘off’ command</td>
</tr>
<tr>
<td>Exit</td>
<td>Internal siren</td>
<td>Stop noise command (after successful entry of PIN)</td>
</tr>
<tr>
<td>Exit</td>
<td>External siren</td>
<td>Start noise command (after three unsuccessful PIN entries, or if the PIN is not entered in time)</td>
</tr>
<tr>
<td>Exit</td>
<td>External siren</td>
<td>Stop noise command (after 20 minutes, a legal requirement)</td>
</tr>
</tbody>
</table>

Size = 9 CFP (COSMIC Function Points)
In summary: COSMIC Function Points

- Designed by an international group of software measurement experts
  - COSMIC: Common Software Measurement International Consortium
- To measure the Functional User Requirements of:
  - Business applications
  - Real-time
  - Infrastructure software
  - Various other types of software
  - Hybrids of these
- Based on:
  - Metrology
  - Fundamental software engineering principles
- An ISO standard: ISO 19761
- Open, freely available (via www.cosmic-sizing.org )
Agenda

1. Software Effort Estimation & Software Size
2. COSMIC: 2\textsuperscript{nd} generation of Function Points
3. Versatility of COSMIC Function Points
4. Contributions of COSMIC to Estimation Models
5. Early & Quick COSMIC sizing at Estimation Time
6. Automation of COSMIC Function Points
7. Summary
Versatility - Guidelines by Application Domains

- Business applications
- Real-time software
- Data Warehouse software
- SOA software (SOA: Service Oriented Architecture)
- Mobile apps
- Agile Development
Versatility - at any level of software requirements

Application Layer

- App 1
- App 2
- App ‘n’

Middleware Layer (Utilities, etc)

Database Management System Layer

- DBMS 1
- DBMS 2

Operating System Layer

- Keyboard Driver
- Screen Driver
- Print Driver
- Disk Driver

Hardware

- Keyboard
- VDU Screen
- Printer
- Hard Disk Drive
- Central Processor

© Copyrights 2017: COSMIC and authors
Versatility - COSMIC Case Studies

- **Real-time:**
  - Rice cooker
  - Automatic line switching
  - Valve control

- **Business:**
  - Course registration (distributed)
  - Restaurant management (web & mobile phone)
  - Banking web advice module
  - Car hire (existing legacy app.)
Agile: Aggregation rules for components, sprints, etc. up to whole software systems

COSMIC size usable for:
- early Total System sizing & effort estimation
- US, Sprint etc. sizing & estimation
- Progress control at any level

© Copyrights 2017: COSMIC and authors
Agenda

1. Software Effort Estimation & Software Size
2. COSMIC: 2nd generation of Function Points
3. Versatility of COSMIC Function Points
4. Contributions of COSMIC to Estimation models
5. Early & Quick COSMIC sizing at estimation time
6. Automation of COSMIC Function Points
7. Summary
COSMIC data from Industry

Practical experimentations with the COSMIC method in Automotive embedded software field

By: Sophie Stern

Renault
Data from Renault - 2012

Engine Control Unit: Modules evolutions, manual coding

Effort (man/day) vs. Functional size = CFP

R^2 = 0.804

© Copyrights Renault 2012
Data from Renault - 2012
Renault: Estimation & Negotiations

BCM RFQ: COSMIC predictions versus Supplier estimations

© Copyrights Renault 2012
Renault - Remarkable cost estimation accuracy from its ECU software specifications

Cost vs size (CFP)

Memory size vs software size (CFP)

© Copyrights Renault 2012

© Copyrights 2017: COSMIC and authors
Industry Data - Example 2: 25 Web applications

Conclusions:
‘The results of the … study revealed that COSMIC outperformed Function Points as indicator of development effort by providing significantly better estimations’

Ref.: ‘Web Effort Estimation: Function Point Analysis vs. COSMIC’
Industry Data - Example 3:
Security & surveillance software systems

- Scrum method
- Teams estimate tasks within each iteration in Story Points
- Measurements of 24 tasks in 9 iterations
  - Each task estimated in Story Points
  - Task actual effort recorded
  - Each task also measured in CFP

Industry Data - Example 3: Security & surveillance software systems

Effort = 0.47 x Story Points + 17.6 hours and $R^2 = 0.33$
Industry Data - Example 3:
Security & surveillance software systems

**Effort = 0.47 x Story Points + 17.6 hours** and $R^2 = 0.33$

**Y = 2.35 x CFP - 0.08hrs** and $R^2 = 0.977$
Industry Data - Example 4: Vector Consulting Group (Germany) Manufacturing, Engineering, Automotive, ..)

COSMIC Benefits

- Agreed model for measuring functional size
  - Solid baseline for benchmarking
- Vector achieved with many clients a precision of 10-20% within one year of building the estimation program:
  - Transparent effort estimations on the basis of functional changes
  - Ad-hoc & fuzzy evaluations and negotiations for single SW changes are reduced
  - Significantly increased efficiency & trust for better collaboration between supplier & customer

© Copyrights 2017: COSMIC and authors
Other sources of COSMIC examples

- COSMIC web site at: www.cosmic-sizing.org
Agenda

1. Software Effort Estimation & Software Size
2. COSMIC: 2nd generation of Function Points
3. Versatility of COSMIC Function Points
4. Contributions of COSMIC to Estimation models
5. Early & Quick COSMIC sizing at estimation time
6. Automation of COSMIC Function Points
7. Summary
Quality of the documentation of a functional process at measurement time

<table>
<thead>
<tr>
<th>Functional Process Quality Level</th>
<th>Quality of the functional process definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completely defined</td>
<td>Functional process and its data movements are completely defined</td>
</tr>
<tr>
<td>Documented</td>
<td>Functional process is documented but not in sufficient detail to identify the data movements</td>
</tr>
<tr>
<td>Identified</td>
<td>Functional process is listed but no details are given of its data movements</td>
</tr>
<tr>
<td>Counted</td>
<td>A count of the functional processes is given, but there are no more details</td>
</tr>
<tr>
<td>Implied (A 'known unknown')</td>
<td>The functional process is implied in the actual requirements but is not explicitly mentioned</td>
</tr>
<tr>
<td>Not mentioned (An 'unknown unknown')</td>
<td>Existence of the functional processes is completely unknown at present</td>
</tr>
</tbody>
</table>
Discuss the applicability, reported use, strengths & weaknesses of 8 approximation techniques:

1. Average functional process approximation
2. Fixed size classification approximation
3. Equal size bands approximation
4. Average use case approximation
5. Early & quick COSMIC approximation
6. Easy function points approximation
7. Approximation from informally written texts
8. Approximation using fuzzy logic
Example 1: Fixed size intervals

<table>
<thead>
<tr>
<th>Classification</th>
<th>Size (CFP)</th>
<th>#E</th>
<th>#X</th>
<th>#R</th>
<th>#W</th>
<th>Error messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Medium</td>
<td>10</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Large</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example 2: Equal size bands

Equal size bands from 37 business applications

<table>
<thead>
<tr>
<th>Band</th>
<th>Average size of a Functional Process</th>
<th>% of total Functional Size</th>
<th>% of total number of Functional Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>4.8</td>
<td>25%</td>
<td>40%</td>
</tr>
<tr>
<td>Medium</td>
<td>7.7</td>
<td>25%</td>
<td>26%</td>
</tr>
<tr>
<td>Large</td>
<td>10.7</td>
<td>25%</td>
<td>19%</td>
</tr>
<tr>
<td>Very Large</td>
<td>16.4</td>
<td>25%</td>
<td>15%</td>
</tr>
</tbody>
</table>

Equal size bands from a major component of an avionics system

<table>
<thead>
<tr>
<th>Band</th>
<th>Average size of a Functional Process</th>
<th>% of total Functional Size</th>
<th>% of total number of Functional Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>5.5</td>
<td>25%</td>
<td>49%</td>
</tr>
<tr>
<td>Medium</td>
<td>10.8</td>
<td>25%</td>
<td>26%</td>
</tr>
<tr>
<td>Large</td>
<td>18.1</td>
<td>25%</td>
<td>16%</td>
</tr>
<tr>
<td>Very Large</td>
<td>38.8</td>
<td>25%</td>
<td>7%</td>
</tr>
</tbody>
</table>
Example 3: Probability distribution in the Business domain

<table>
<thead>
<tr>
<th>Classification of the FP</th>
<th>Specification level</th>
<th>CFP (min)</th>
<th>CFP (mean)</th>
<th>CFP (max)</th>
<th>Approximate CFP</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small FP</td>
<td>Little unknown</td>
<td>2 (10%)</td>
<td>3 (75%)</td>
<td>5 (15%)</td>
<td>3.2</td>
<td>&gt;80%</td>
</tr>
<tr>
<td>Small FP</td>
<td>Unknown (No FUR)</td>
<td>2 (15%)</td>
<td>4 (50%)</td>
<td>8 (35%)</td>
<td>5.1</td>
<td>&lt;50%</td>
</tr>
<tr>
<td>Medium FP</td>
<td>Little unknown</td>
<td>5 (10%)</td>
<td>7 (75%)</td>
<td>10 (15%)</td>
<td>7.25</td>
<td>&gt;80%</td>
</tr>
<tr>
<td>Medium FP</td>
<td>Unknown (No FUR)</td>
<td>5 (15%)</td>
<td>8 (50%)</td>
<td>12 (35%)</td>
<td>8.95</td>
<td>&lt;50%</td>
</tr>
<tr>
<td>Large FP</td>
<td>Little unknown</td>
<td>8 (10%)</td>
<td>10 (75%)</td>
<td>12 (15%)</td>
<td>10.1</td>
<td>&gt;80%</td>
</tr>
<tr>
<td>Large FP</td>
<td>Unknown (No FUR)</td>
<td>8 (15%)</td>
<td>10 (50%)</td>
<td>15 (35%)</td>
<td>11.45</td>
<td>&lt;50%</td>
</tr>
<tr>
<td>Complex FP</td>
<td>Little unknown</td>
<td>10 (10%)</td>
<td>15 (75%)</td>
<td>20 (15%)</td>
<td>15.25</td>
<td>&gt;80%</td>
</tr>
<tr>
<td>Complex FP</td>
<td>Unknown (No FUR)</td>
<td>10 (15%)</td>
<td>18 (50%)</td>
<td>30 (35%)</td>
<td>21</td>
<td>&lt;50%</td>
</tr>
</tbody>
</table>
Agenda

1. Software Effort Estimation & Software Size
2. COSMIC: 2nd generation of Function Points
3. Versatility of COSMIC Function Points
4. Contributions of COSMIC to Estimation Models
5. Early & Quick COSMIC sizing at estimation time
6. Automation of COSMIC Function Points
7. Summary
COSMIC specifications for Automation with Matlab-Simulink
Map the Graph Notation to COSMIC Model of Software

TABLE I.

<table>
<thead>
<tr>
<th>COSMIC concepts</th>
<th>COSMIC abbreviation</th>
<th>Proposed graphical representation</th>
<th>Proposed graphical description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functional user</td>
<td>FU</td>
<td></td>
<td>Green dashed box</td>
</tr>
<tr>
<td>Functional process</td>
<td>FP</td>
<td></td>
<td>Blue box</td>
</tr>
<tr>
<td>Data group movement</td>
<td>E/X/W/R</td>
<td></td>
<td>Black arrow</td>
</tr>
<tr>
<td>Persistent storage</td>
<td></td>
<td></td>
<td>ISO 5807 stored data symbol in light blue</td>
</tr>
</tbody>
</table>

AUTOMATION ACCURACY REACHED WITH COSMIC

Steer-by-wire case study

<table>
<thead>
<tr>
<th>Steer-by-Wire Runnable</th>
<th>Functional size obtained by the manual FSM procedure (CFP)</th>
<th>Functional size obtained by the automated FSM procedure (CFP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steer_Run_Acquisition</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Steer_Run_Sensor</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Steer_Run_Command</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Steer_InterECU_Wheel</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Steer_Run_Actuator</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Wheel_Run_Acquisition</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Wheel_Run_Sensor</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Wheel_Run_Command</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Wheel_InterECU_Steer</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Wheel_Run_Actuator</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>38</strong></td>
<td><strong>38</strong></td>
</tr>
</tbody>
</table>

Automation in Industry

<table>
<thead>
<tr>
<th>Total Number of Models</th>
<th>Total Size obtained manually (CFP)</th>
<th>Total Size obtained using the prototype tool (CFP)</th>
<th>Difference (%)</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>All 77 models</td>
<td>1,758</td>
<td>1,791</td>
<td>1.8%</td>
<td>&gt;98%</td>
</tr>
<tr>
<td>76 fault-free models</td>
<td>1,729</td>
<td>1,739</td>
<td>Less than 1%</td>
<td>&gt;99%</td>
</tr>
</tbody>
</table>


© Copyrights 2017: COSMIC and authors
Protocol for Verifying the Accuracy of Automation

**Phase 1**
- **Do the final results match?**
  - yes → Continue verifying the measurement in detail?
    - yes → End of high-level verification
    - no → End of detailed verification
  - no → Phase 2

**Phase 2**
- **Same number of FPs?**
  - yes → Continue verifying the measurement in detail?
    - yes → End of high-level verification
    - no → Inspection of the quality of the specification
  - no → **Same identified FPs in both measures?**
    - yes → Continue verifying the measurement in detail?
      - yes → End of high-level verification
      - no → Inspection of the tool's modules
    - no → **Same data movements in both measures?**
      - yes → End of detailed verification
      - no → Phase 3

**Phase 3**
- **Difference caused by human error?**
  - yes → End of detailed verification
  - no → Defect/quality or bug report

© Copyrights 2017: COSMIC and authors

ICEAA Portland - Oregon (USA), June 2017
COSMIC Automation in SCADE

- Scade: A safety-certified language
  https://www.youtube.com/watch?v=gjCvOjaCY88

- https://www.ijerst.com/ijerstadmin/upload/IJEETC_554b274b6329d.pdf
Agenda

1. Software Effort Estimation & Software Size
2. COSMIC: 2nd Generation of Function Points
3. Versatility of COSMIC Function Points
4. Contributions of COSMIC to Estimation Models
5. Early & Quick COSMIC sizing at Estimation Time
6. Automation & Accuracy of COSMIC Function Points
7. Summary
The COSMIC method is used various countries

- COSMIC Measurement Manual standard (11 languages)
- Size of user base is unknown
  - Of known users, 50% are software houses
  - Adopted by Governments (Mexico, Poland, China...)
  - > 30,000 downloads of research & conference papers
- USA: GAO\(^1\), NIST\(^2\) documents
- + 600 certification exam holders (ex. China, India, Mexico, Italy, Poland, Turkey, Brazil)
- Two active forums (on Linkedin CUG, [www.cosmic-sizing.org](http://www.cosmic-sizing.org))

Summary of benefits

- Free & open
- Fundamental SE Principles: future-proof, stable
- Very wide applicability across software domains & layers
- Proven value for performance measurement & estimating
- ISO standard
- Can be automated with very high accuracy & traceability
Conclusion

Software COST Estimating: critical knowledge for today and tomorrow

Ample industry evidence that COSMIC Function Points allow:

- Meaningful benchmarking
- Estimation with very low variations (... conditions apply...)
- Automation with high precision
Acknowledgements

The authors wish to acknowledge the efforts of members of the COSMIC Measurement Practices Committee and many others who, over the last 18 years, have contributed to the development and implementation of the COSMIC method.
Thank you for your attention

(www.cosmic-sizing.org)

Alain Abran  alain.abran@etsmtl.ca
Charles Symons  cr.symons@btinternet.com
Christof Ebert  christof.ebert@vector.com
Frank Vogelezang  frank.Vogelezang@ordina.nl
Hassan Soubra:  hassan.soubra@estaca.fr
There is a well-defined Measurement Process

**Phase 1**
**Measurement Strategy**
- Input from measurement sponsor
- Software Context Model
- Functional User Requirements (FUR)

**Phase 2**
**Mapping Phase**
- Definition of each piece of software to be measured and of the required measurement (Purpose and scope)
- COSMIC Generic Software Model
- FUR in the form of the COSMIC Generic Software Model

**Phase 3**
**Measurement Phase**
- Functional size of the software in units of CFP

© Copyrights 2017: COSMIC and authors

ICEAA Portland - Oregon (USA), June 2017
What to do about NFR?

Again, there was no good standard definition of a NFR.

A joint COSMIC/IFPUG effort developed good definitions and a Glossary of NFR and Project Requirements.

The COSMIC Guideline advises how to deal with NFR.

Glossary of terms for Non-Functional Requirements and Project Requirements used in software project performance measurement, benchmarking and estimating

Version 1.0
September 2015

Guideline on Non-Functional & Project Requirements

How to consider non-functional and project requirements in software project performance measurement, benchmarking and estimating

Version 1.
November 2015
Abran & Al Sarayreh showed that requirements that appear as NFR may evolve into FUR, that the COSMIC method can measure.
Examples of NFR leading to FUR with COSMIC

A Standards-Based Model of System Maintainability Requirements

Khaled T. Al-Atrash

Software Engineering Department, Rocks University, Alexandria, Egypt

Alaa Al Nhiều

Software Engineering Department, University of Quebec (CT), 1109 Sainte-Diene, Chicoutimi, Quebec, Canada

Juan J. Cuenca-Colon

Department of Computer Science, Universidad de Alcalá, 28850 Alcalá de Henares, Madrid, Spain

1. SUMMARY

The non-functional requirements are often captured only generically at a fairly high level, and they do not include the levels of detail necessary at this stage for the system engineers to allocate them as specific maintainability to be handled either by the software or the hardware, or a specific combination of the two. The European EC55 series of standards for the aeronautic industry includes maintainability requirements in one of its types of non-functional requirements (NFR) for embedded and real-time software. A number of maintainability-related concepts are dispersed throughout the EC55 ISO 9126, and IEEE standards to describe, at varying levels of detail, the various types of maintainability requirements in the system, software, and hardware levels. This paper organizes these dispersed maintainability concepts into a standards-based reference model of system maintainability requirements. The availability of this reference model can facilitate the early identification of the system maintainability-NFR and their detailed allocation as specific maintainability functions to be handled by the allocated allocation to hardware or software, or a specific combination of the two. In the absence of such a reference model, these NFR requirements are typically handled in practice much later on in the software development life cycle, when, at system testing time, users and developers find out that a number of maintainability requirements have been overlooked and additional effort has to be expended to implement them. The approach adopted in this research for the structure of this reference NFR model is based on the genetic model of software functional requirements proposed in the COSMIC -- ISO 16755 model, thereby allowing the functional size of each maintainability requirement allocated to software to be measured.


* Correspondence to: Khaled T. Al-Atrash, Software Engineering Department, Rocks University (2017), 1154 Zara, Boston, ME email: khaled@rocks.edu"
Example 2: with a Message Sequence Diagram

- **Boundary**
  - FP of App X being measured
  - Triggering: E
  - Another: E
- **Human Functional User**
  - Item detail: X
  - Total: X
  - Error msg.: X
- **Boundary**
  - FP of Software Functional User of App X
  - Message to the other software: E
  - Reply from the other software: X
  - R (for validation)

Size = 9 CFP
Guidelines for Practitioners

A Guideline describing a range of Approximate Sizing methods
Size/Cost estimates are usually needed before the FUR have been defined in detail

A Guideline on ‘Assuring the accuracy of COSMIC measurements’
# 1st Generation of Function Points: Weights

<table>
<thead>
<tr>
<th>FTR's</th>
<th>DATA ELEMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>5-15 &gt; 15</td>
</tr>
<tr>
<td>0-1</td>
<td>Low Low Ave</td>
</tr>
<tr>
<td>2</td>
<td>Low Ave High</td>
</tr>
<tr>
<td>&gt; 3</td>
<td>Ave High High</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FTR's</th>
<th>DATA ELEMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>6-19 &gt; 19</td>
</tr>
<tr>
<td>0-1</td>
<td>Low Low Ave</td>
</tr>
<tr>
<td>2-3</td>
<td>Low Ave High</td>
</tr>
<tr>
<td>&gt; 3</td>
<td>Ave High High</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rating</th>
<th>VALUES</th>
</tr>
</thead>
<tbody>
<tr>
<td>EO</td>
<td>EQ</td>
</tr>
<tr>
<td>Low</td>
<td>4</td>
</tr>
<tr>
<td>Average</td>
<td>5</td>
</tr>
<tr>
<td>High</td>
<td>7</td>
</tr>
</tbody>
</table>

**Inputs - Matrix**

**Output & Enquiries - Shared Matrix**

**Transactions: weights in FP**

<table>
<thead>
<tr>
<th>RET's</th>
<th>DATA ELEMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-9</td>
<td>20-50 &gt; 50</td>
</tr>
<tr>
<td>1</td>
<td>Low Low Ave</td>
</tr>
<tr>
<td>2-5</td>
<td>Low Ave High</td>
</tr>
<tr>
<td>&gt; 5</td>
<td>Ave High High</td>
</tr>
</tbody>
</table>

**Files (internal & external) Matrix**

**Files: weights in FP**

© Copyrights 2017: COSMIC and authors

ICEAA Portland - Oregon (USA), June 2017