Do Specification Characteristics Correlate with Delivered Software Size?

Richard D. Stutzke (SAIC)
and
Terry L. Palmer
(Davidson Technology, Inc.)

21 October 2003


Richard.D.Stutzke@saic.com
Motivation

• Requirements determine total product size
• A written specification usually provides the first “good” requirements
• Does the delivered size (SLOC) correlate with information in the specification?
Approach

1. Collect data from completed projects
   - Specification (Microsoft Word)
   - Total size of the delivered code (PSLOC)

2. Measure information in specification
   - Microsoft Word (style checker)
   - Automated Requirements Measurement (ARM) tool*

3. Look for correlations
   - Size
   - Document quality

*from the NASA GSFC Software Assurance Technology Center.
See http://satc.nasa.gov/library
“Good” Strings

• Imperative – word and phrases that command that something be provided
  – “Shall” normally dictates a functional capability
  – “Must”/”must not” establishes performance requirements/constraints
  – “Are applicable” includes, by reference, standards and other documents
  – “Will” indicates something provided from outside the specified product
  – An explicit specification usually has large numbers of imperative words

• Continuances – phrases that follow an imperative and precede low level requirements
  – Examples: “following:”, “in particular:”, and “below:”
  – Continuances indicate how well requirements are organized and structured
  – Many continuances may indicate multiple, overly complex requirements

• Directives – point to illustrative information to make requirements more understandable
  – Examples: “Figure”, “Table”, and “for example”
  – Many directives may indicate more precision in the requirements
“Bad” Strings

• Weak phrases – clauses that are open to multiple interpretations
  – Examples are “adequate” and “as appropriate”
  – May indicate that the requirement is defined elsewhere
  – May indicate that the requirement is open to subjective interpretation

• Options – words that give the developer latitude in implementation of the requirement
  – Examples are “can”, “may”, and “optionally”
  – Options loosen the specification, reducing control over the produce

• Incomplete – words that clearly indicate missing information
  – Examples are “TBD” (To Be Determined) and “TBS” (To Be Supplied)
  – There indicate the document may not contain all the requirements
Other Measures

• Lines of text – the number of individual lines read by the program from the source file
  – Provides a size measure to normalize the data
• Structure depth – count of the numbered statements at each
  – Indicates the document’s organization, consistency, and level of detail
  – High-level specifications are usually a structural depth of four or less
  – Detailed documents may have depth up to nine or so
  – A well structured document with consistent levels of detail has pyramidal shape
• Specification depth – the number of imperatives at each level of the document
  – Includes count of low level list items that are introduced at a higher level by an imperative followed by a continuance
  – It reflects the structure of the actual requirements
• Comments
  – Differences between the shape of the numbering and specification structures indicate the amount of introductory or background information in the document
  – The ratio of the total for the specification structure (the sum for all levels) to the total lines of text indicates how concise the document is
Document Size Distribution

![Document Size Distribution Chart]
Requirements Per Page

\[ y = 6.4373x \]
\[ R^2 = 0.8199 \]
Requirements versus Strings

\[ y = 0.1428x \]

\[ R^2 = 0.9066 \]
Requirements versus Sentences

\[ y = 0.7593x \]

\[ R^2 = 0.983 \]
Delivered Size versus Requirements
Size versus Requirements

\[ y = 0.4584x \]
\[ R^2 = -1.9464 \]

\[ y = 0.2015x \]
\[ R^2 = 0.9118 \]

\[ y = 0.0001x^2 + 0.0405x \]
\[ R^2 = 0.9924 \]
Prediction Accuracy (Detailed Spec)

<table>
<thead>
<tr>
<th>Project</th>
<th>Type</th>
<th># Reqmts</th>
<th>Size (del)</th>
<th>Size(est)</th>
<th>Error</th>
<th>MRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>D2</td>
<td>ND</td>
<td>70</td>
<td>1.5</td>
<td>14.1</td>
<td>13</td>
<td>840</td>
</tr>
<tr>
<td>D3</td>
<td>ND</td>
<td>328</td>
<td>29.2</td>
<td>66.1</td>
<td>37</td>
<td>126</td>
</tr>
<tr>
<td>D4</td>
<td>ND</td>
<td>95</td>
<td>7.0</td>
<td>19.1</td>
<td>12</td>
<td>173</td>
</tr>
<tr>
<td>D5</td>
<td>ND</td>
<td>70</td>
<td>8.0</td>
<td>14.1</td>
<td>6</td>
<td>76</td>
</tr>
<tr>
<td>D6</td>
<td>ND</td>
<td>50</td>
<td>1.0</td>
<td>10.1</td>
<td>9</td>
<td>908</td>
</tr>
<tr>
<td>D27</td>
<td>SU</td>
<td>85</td>
<td>1.4</td>
<td>17.1</td>
<td>16</td>
<td>1123</td>
</tr>
<tr>
<td>D28</td>
<td>SU</td>
<td>69</td>
<td>1.8</td>
<td>13.9</td>
<td>12</td>
<td>672</td>
</tr>
<tr>
<td>D32</td>
<td>SU</td>
<td>1407</td>
<td>325.0</td>
<td>283.5</td>
<td>-41</td>
<td>13</td>
</tr>
<tr>
<td>D39</td>
<td>SU</td>
<td>325</td>
<td>6.2</td>
<td>65.5</td>
<td>59</td>
<td>956</td>
</tr>
<tr>
<td>D42</td>
<td>SU</td>
<td>709</td>
<td>112.0</td>
<td>142.9</td>
<td>31</td>
<td>28</td>
</tr>
</tbody>
</table>

\[ a = 0.2015 \]

\[ \text{MMRE} = 492 \]

\[ \text{PRED(25)} = 0.1 \]

\[ \text{RMS} = 29 \text{ KSLOC} \]
Levels versus Pages

\[ y = 0.0306x + 3.2192 \]
\[ R^2 = 0.53 \]

\[ y = 0.0348x + 2.8658 \]
\[ R^2 = 0.7027 \]
Requirements Ratio versus Pages

![scatter plot showing requirements ratio versus pages]

- Detailed
- Top Level

# Pages

0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

0 20 40 60 80 100 120 140 160 180 200

Ratio

(requirements)
Other Factors

• Application Complexity
• Choice of Programming Language
  – Six HOLs used in our sample
  – Some products written in multiple HOLs
• Choice of Platform
  – Operating System services
  – COTS Middleware
Conclusions

• Data suggests software size may be predictable from number of requirements
  – Our requirements include growth
  – Effect of document quality not studied

• Level of detail in specification needs to be better defined

• Will need to adjust size for:
  – Programming language
  – Platform
  – Growth
  – Volatility