Software Quality

And Human Behaviour
My Training

• Mathematics
  • Applied, Quantum Mechanics
  • A long time ago
• Meteorology
• Astronomy
• = Computation
  • The more the merrier
My Night Job

User: administrator
2015-11-19 04:55:33
Telescope is idle

The roof has been closed because: *** THE SUN IS (COMING) UP ***
My Night Job

- Optical
- Radio
My Day Job

IBM
But I sometimes work with “Big Data”

• Smart Water Meters
  • Measure every hour versus once a quarter
  • Drive behaviour through information

• Vivid Sydney
  • Tracking people movements using their smartphones

• SKA
  • Gathering data at TBps
  • How do you process it?
  • What do you keep?
I work in Pre-Sales, but occasionally...
I get my hands dirty
I walked into a room...

• 40 people
  • Not happy

• More on the phone

• Problem had been running for 3 days

• No one knows where the problem is
  • Or where to look

• Turned out to be a software problem
  • Fixed 3 years ago
Turned out to be a HIPER

• Highly pervasive problem
  • Customers are notified that the problem exists
  • And that a PTF = fix is available
• The HIPER PTF was PE = “fix in error”
• The fix to the fix was also PE
• Are HIPER PTFs more likely to be PE?
• How long should you leave a HIPER PTF before applying it?
Are HIPER PTFs more likely to be PE

• Yes! (confirmed by hypothesis testing)
• Across several products (thousands of PTFs)
  • ~ 10% of regular PTFs are PE
  • ~ 15% of HIPER PTFs are PE
• The QA process is the same for both
• The pressure is not the same!
• The product teams who publish their bugs more openly performed better 😊
How is Software Quality Measured?
<table>
<thead>
<tr>
<th>Semester in analysis: Jun 2011- Nov 2011</th>
<th>Previous semester: Dec 2010 - May 2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total PMR</td>
<td>205 PMR (Decreased 3% in this semester)</td>
</tr>
<tr>
<td></td>
<td>v1: 49%</td>
</tr>
<tr>
<td></td>
<td>v2: 42%</td>
</tr>
<tr>
<td></td>
<td>v3: 3%</td>
</tr>
<tr>
<td></td>
<td>Other: 5%</td>
</tr>
<tr>
<td></td>
<td>211 PMR</td>
</tr>
<tr>
<td>Customer</td>
<td>118 Customer (decreased 1% in this semester)</td>
</tr>
<tr>
<td></td>
<td>119 Customer</td>
</tr>
<tr>
<td>Sev 1 PMR</td>
<td>1 PMR (0.4% of the total) decreased 74% in this semester</td>
</tr>
<tr>
<td></td>
<td>4 PMR (1.9% of the total)</td>
</tr>
<tr>
<td>PMR/Customer</td>
<td>1.73 (Decreased 2% in this semester)</td>
</tr>
<tr>
<td></td>
<td>1.77</td>
</tr>
</tbody>
</table>
How is Software Quality Measured?

• Number of PMRs
• Time from PMR to APAR
  • Analysis became less complete
• Time from APAR to PTF
  • More PEs
• No special focus on HIPERs or Pes
• Measurement consistently drives behaviour
How long should you leave a HIPER PTF?

- ~90% of HIPER PTFs that turn out to be PE are found to be in error within 60 days
- But I can’t tell anybody 😊
- So, just do it, otherwise we never find the bugs!
Conclusion

• The “Observer Effect” does act on software quality
• Carefully consider how you measure it
  • Will it drive the right outcomes?
• Process does not guarantee consistency
  • Humans will find ways around it
• Never trust what you read/hear
  • Always check the data