COTS Empirical Data: A Top-10 List And Its Implications

Barry Boehm, USC and CeBASE*
Opening Address, GSAW 2002
March 13, 2002

(boehm@sunset.usc.edu)
(http://www.cebase.org)
*Center for Empirically-Based Software Engineering
Outline

- COTS Software: A Behavioral Definition
- COTS Empirical Hypotheses: Top-10 List
  - COTS Best Practice Implications
- COTS-Based Systems (CBS) Challenges
COTS Software: A Behavioral Definition

- Commercially sold and supported
  - Avoids expensive development and support
- No access to source code
  - Special case: commercial open-source
- Vendor controls evolution
  - Vendor motivation to add, evolve features
- No vendor guarantees
  - Of dependability, interoperability, continuity, ...
COTS Top-10 Empirical Hypotheses-I
- Basili & Boehm, *Computer*, May 2001, pp. 91-93

- Over 99% of all executing computer instructions come from COTS
  - Each has passed a market test for value
- More than half of large-COTS-product features go unused
- New COTS release frequency averages 8-9 months
  - Average of 3 releases before becoming unsupported
- CBS life-cycle effort can scale as high as $N^2$
  - $N = \#$ of independent COTS products in a system
- CBS post-deployment costs exceed development costs
  - Except for short-lifetime systems
Usual Hardware-Software Trend Comparison

- Different counting rules
- Try counting software as Lines of Code in Service (LOCS)
  \[ = \sum \text{(#platforms)} \times \text{(#object LOCS/platform)} \]

[Graph showing the relationship between Lines of Code (LOCS) and Total $/LOCS from 1950 to 2000.]
1. DoD LOCS: % COTS, 2000

M = Million, T = Trillion

<table>
<thead>
<tr>
<th>Platform</th>
<th># P’forms (M)</th>
<th>LOCS P’forms (M)</th>
<th>LOCS (T)</th>
<th>% COTS</th>
<th>Non-COTS LOCS (T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mainframe</td>
<td>.015</td>
<td>200</td>
<td>3</td>
<td>95</td>
<td>.15</td>
</tr>
<tr>
<td>Mini</td>
<td>.10</td>
<td>100</td>
<td>10</td>
<td>99</td>
<td>.10</td>
</tr>
<tr>
<td>Micro</td>
<td>2</td>
<td>100</td>
<td>200</td>
<td>99.5</td>
<td>1.00</td>
</tr>
<tr>
<td>Combat</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>80</td>
<td>.80</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>217</td>
<td></td>
<td></td>
<td>2.05</td>
</tr>
</tbody>
</table>

(<1%)  

- COTS often an economic necessity
2. Use of COTS Features: Microsoft Word and Power Point
- K. Torii Lab, Japan: ISERN 2000 Workshop

- Individuals: 12-16% of features used
- 10-Person Group: 26-29% of features used

- Extra features cause extra work
- Consider build vs. buy for small # features
3. COTS Release Frequency: Survey Data
   - Ron Kohl surveys: GSAW 1999-2001*

<table>
<thead>
<tr>
<th>GSAW Survey</th>
<th>Release Frequency (months)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1999</td>
<td>6.3</td>
</tr>
<tr>
<td>2000</td>
<td>8.5</td>
</tr>
<tr>
<td>2001</td>
<td>8.75</td>
</tr>
</tbody>
</table>

- Adaptive maintenance often biggest CBS life cycle cost
- Average of 3 releases before becoming unsupported

4. CBS Effort Scaling As High As $N^2$

$N = \text{number of independent COTS products}$

**Strong COTS Coupling**

- COTS
- COTS
- Custom SW

Effort $\sim N + N(N-1)/2 = N(N+1)/2$ (here = 6)

**Weak COTS Coupling**

- COTS
- Custom SW

Effort $\sim N$ (here = 3)
4. CBS Effort Scaling - II

- Reduce # COTS or Weaken COTS coupling
  - COTS choices, wrappers, domain architectures, open standards, COTS refresh synchronization
5. CBS Life Cycle Costs: The COTS-LIMO Model

- Chris Abts, ESCOM 2001

Cost of maintenance
Fn (synchronization, complexity of system, no. planned upgrades, etc.)

Volatility effects dominate increased integration experience

Increased integration experience dominates volatility effects

Volatility effects just cancel increased integration experience

Retire
Maintain
Time

No. of COTS in system

n+X
n+3
n+2
n+1
n
5
4
3
2
1

$
COTS Top-10 Empirical Hypotheses-II

- Less than half of CBS development effort comes from glue code
  - But glue code costs 3x more per instruction
- Non-development CBS costs are significant
  - Worth addressing, but not overaddressing
- COTS assessment and tailoring costs vary by COTS product class
- Personnel factors are the leading CBS effort drivers
  - Different skill factors are needed for CBS and custom software
- CBS project failure rates are higher than for custom software
  - But CBS benefit rates are higher also
6. COCOTS Effort Distribution: 20 Projects

- Mean % of Total COTS Effort by Activity ( +/- 1 SD)

- Glue code generally less than half of total
- No standard CBS effort distribution profile
7. Non-Development CBS Costs Are Significant

- Include licenses, training, vendor relations
- Benefits of addressing them
  - Volume discounts: $1-10M/year savings
  - Unified vendor interface
- Costs of overaddressing them
  - Go with immature open-source freeware
  - Using COTS license cost as COTS assessment cost limit

- Use risk to determine how much COTS assessment is enough
8. Median Assessment Effort by COTS Class

- COCOTS database: 20 projects

Person-months

- DBMS, network COTS most likely sources of assessment effort
- Again, no standard CBS effort distribution
8. Median Tailoring Effort by COTS Class

- COCOTS database: 20 projects

- Operating systems: 2.00 person-months
- Network managers: 12.67 person-months
- GUI: 14.00 person-months
- DBMS: 38.29 person-months

- DBMS the most likely source of COTS tailoring effort
- Again, no standard CBS effort distribution

03/13/02 ©USC-CSE
9. Personnel Factors Are Leading CBS Effort Drivers: Different Skill Factors Needed

<table>
<thead>
<tr>
<th>Custom Development</th>
<th>CBS Development</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Rqts./implementation assessment</td>
<td>• Rqts./ COTS assessment</td>
</tr>
<tr>
<td>• Algorithms; data &amp; control structures</td>
<td>• COTS mismatches; connector structures</td>
</tr>
<tr>
<td>• Code reading and writing</td>
<td>• COTS mismatches; assessment, tailoring, glue code development; coding avoidance</td>
</tr>
<tr>
<td>• Adaptation to rqts. changes</td>
<td>• Adaptation to rqts., COTS changes</td>
</tr>
<tr>
<td>• White-box/black-box testing</td>
<td>• Black-box testing</td>
</tr>
</tbody>
</table>

• Rethink personnel recruiting and evaluation criteria
10. CBS Projects Tend to Fail Hard

- Major Sources of CBS Project Failure

  - CBS skill mismatches
  - CBS inexperience
  - CBS optimism
  - Weak life cycle planning
  - CBS product mismatches
  - Hasty COTS assessment
  - Changing vendor priorities
  - New COTS market entries

  • These are major topics for COTS risk assessment
CBS Challenges

- Process specifics
  - Milestones; role of “requirements”
  - Multi-COTS refresh frequency and process
- Life cycle management
  - Progress metrics; development/support roles
- Cost-effective COTS assessment and test aids
  - COTS mismatch detection and redressal
- Increasing CBS controllability
  - Technology watch; wrappers; vendor win-win
- Better empirical data and organized experience