Carnegie Mellon
Software Engineering Institute

Pittsburgh, PA 15213-3890

Improving Acquisition through COTS Risk Identification

David Carney, Ed Morris, Pat Place

Sponsored by the U.S. Department of Defense

Carnegie Mellon
Software Engineering Institute

Motivation

Inform the community of results seen through CURE application

Provide recommendations to improve acquisition of COTS-based systems

Carnegie Mellon
Software Engineering Institute

Outline

- CURE history

Outline of CURE method

Risks identified by CURE

Recommendations

Carnegie Mellon
Software Engineering Institute

Motivation for CURE

The SEI has been called in to assess many programs

A common factor in many assessments was the presence of COTS software

Similar situations meant we were asking the same questions in every assessment about COTS software

Idea was to ask the questions early a program's life cycle, **before** the risks became problems

Carnegie Mellon
Software Engineering Institute

CURE chronology

FY99:
- Began development of questionnaire
- Developed initial COTS risk list
- Piloted CURE on a small program

FY00:
- Created CURE method
- Created database-supported analysis approach

FY01:
- Stabilized analysis method

FY02 & FY03:
- Developed Discussion Document from questionnaire
- Created Evaluation Record
- Refined and harmonized all artifacts
- Created Reference Manual

Carnegie Mellon
Software Engineering Institute

Outline

CURE history

- Outline of CURE method

Risks identified by CURE

Recommendations

Carnegie Mellon
Software Engineering Institute

High-level overview

CURE method consists of:

- an initial questionnaire
- an on-site interview
- a period of data analysis
- an outbrief

Initial questionnaire is sent 4 weeks in advance of the interview

Interview lasts 1-1½ days

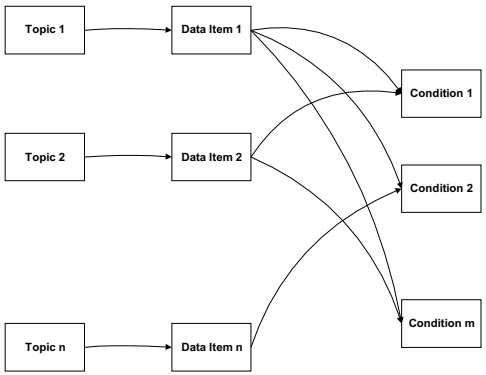
Outbrief is given no later than 4 days after interview

Anatomy of a risk

Risks comprise a condition and a consequence

- conditions can be known in advance (observed in other programs)
- consequences are dependent on the program

The existence of a condition is dependent on many pieces of data



Carnegie Mellon
Software Engineering Institute

Interview and analysis

Interview:

- starts with an inbrief that describes
  - CURE to the participants
  - the program
- raises all topics in the Discussion Document
- evaluators fill in data items
- is not adversarial

Analysis:

- evaluators agree on data items
- generate report of conditions and associated data
- determine whether conditions apply
- develop consequences and mitigations
- prepare outbrief (typically about a dozen risks)

Carnegie Mellon
Software Engineering Institute

Outline

CURE history

Outline of CURE method

- Risks identified by CURE

Recommendations

Carnegie Mellon
Software Engineering Institute

CURE results

Data collected from CUREs performed on DoD and other government agencies

Programs included:

- weapons systems
- information systems
- command, control, communication, and intelligence (C3I) systems

Data rolled up into 4 categories

Carnegie Mellon
Software Engineering Institute

Programmatics and Management

Accounted for 31% of identified risks

- Personnel (7%)
  - Shifts in the Program Office will cause disruptions
- Budget (5%)
- Requirements management (4%)
- Contract (4%)
- Risk management (3%)
  - Diffuse risk management process will lead to risks being overlooked
- Decision making (3%)
- IPTs (3%)
  - Management team lacks experience with IPTs making them a hindrance rather than a benefit
- Process (2%)

Carnegie Mellon
Software Engineering Institute

Mission and Stakeholders

Accounted for 30% of identified risks

- Fulfilling system requirements (12%)
  - Performance requirements will not be met
- Sustainment (9%)
  - The plans and responsibilities for long-term maintenance are undefined
- Stakeholders (9%)
  - The users will find the new business processes unacceptable

Carnegie Mellon
Software Engineering Institute

Technical Areas

Accounted for 24% of identified risks

- Business processes (5%)
  - The business or technical processes embedded in the completed system will not match all stakeholder expectations
- Configuration management (5%)
- Fielding (4%)
  - Lack of attention to installation details will cause fielding difficulties
- Test (4%)
- Integration (3%)
- Data conversion (2%)
- Evaluation (1%)

Carnegie Mellon
Software Engineering Institute

Product-Specific Issues

Accounted for 16% of identified risks

- COTS modification (6%)
  - Estimates about modification of NDI are overly optimistic
- COTS version skew (5%)
  - Gap between the system specific and commercial versions of key commercial component x will become significant
- COTS product support (5%)
  - Key commercial component x will not become a mature and viable commercial product

Carnegie Mellon
Software Engineering Institute

Analysis

Product-specific issues were relatively few – perhaps because the questions are more management oriented

Different profile from Tri-Service Assessment Initiative

- COTS focus of CURE
- different choices on data roll-up

Carnegie Mellon
Software Engineering Institute

Outline

CURE history

Outline of CURE method

Risks identified by CURE

- Recommendations

Carnegie Mellon
Software Engineering Institute

Educate appropriately

Many of the risks we identified are easily avoided

Verified the need for COTS skills in acquisition

NDI behaves a lot like COTS software

Add/strengthen COTS software components – particularly in area of setting realizable requirements (single biggest sub-category)

Carnegie Mellon
Software Engineering Institute

Extend CURE

CURE probe is “painless, quick, and productive”

But, even with the COTS focus of CURE, we uncovered other, non-COTS risks

Differences between contractor and program office community are striking – early discovery of differences reduces risk

- always interview government and contractor

Do follow-ups

Carnegie Mellon
Software Engineering Institute

Rethink attitude toward risk

CURE was hard to even give away
- takes time of senior program staff
- fear of risks

Need to change the mindset

risk ≠ problem

Manage by risk
- identify all plausible risks
- mitigate as appropriate

Carnegie Mellon
Software Engineering Institute

For more information

David Carney, Ed Morris, and Pat Place
(412) 268-7746
cure@sei.cmu.edu

Also “Identifying Commercial Off-the-Shelf (COTS) Product Risks: The COTS Usage Risk Evaluation”
CMU/SEI-2003-TR-023 ESC-TR-2003-03
http://www.sei.cmu.edu/publications/documents/03.reports/03tr023.html