

COTS Product Selection

T. Scott Ankrum

The **MITRE** Corporation

June 22, 2010

Why a Formal Process is Needed

- Selection is usually more complex than expected
- Many candidate products are often considered
- Consider other criteria besides product functions
- The decision needs to be justified.

The Basis and Origin of This Work

- Ankrum, T.S, Cook, R, Qunhui , N. (2000). “CASM Version Control Product Selection” unpublished project at the University of Maryland University College.
 - The original work that developed the major improvements over most other processes.
- Oberndorf, T., Comella-Dorda, S., Dean, J., & Morris, E. (2000). “Picking the Right COTS Product.” Proceedings of the Software Engineering Institute Symposium 2000.
 - Inspired the original work

Related Selection Process Work

- Bandor, M. (2006). “Quantitative Methods for Software Selection and Evaluation” available: <http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA460422>.
- MITRE Corporation. (2010). “Standardized Technology Evaluation Process” (STEP). Internal to MITRE.
 - Maintained by the Systems Engineering Practice Office (SEPO)
- Space and Naval Warfare Systems Center. (2002). “Commercial-Off-the-Shelf (COTS) Evaluation, Selection, and Qualification Process” available: <http://sepo.spawar.navy.mil/>.

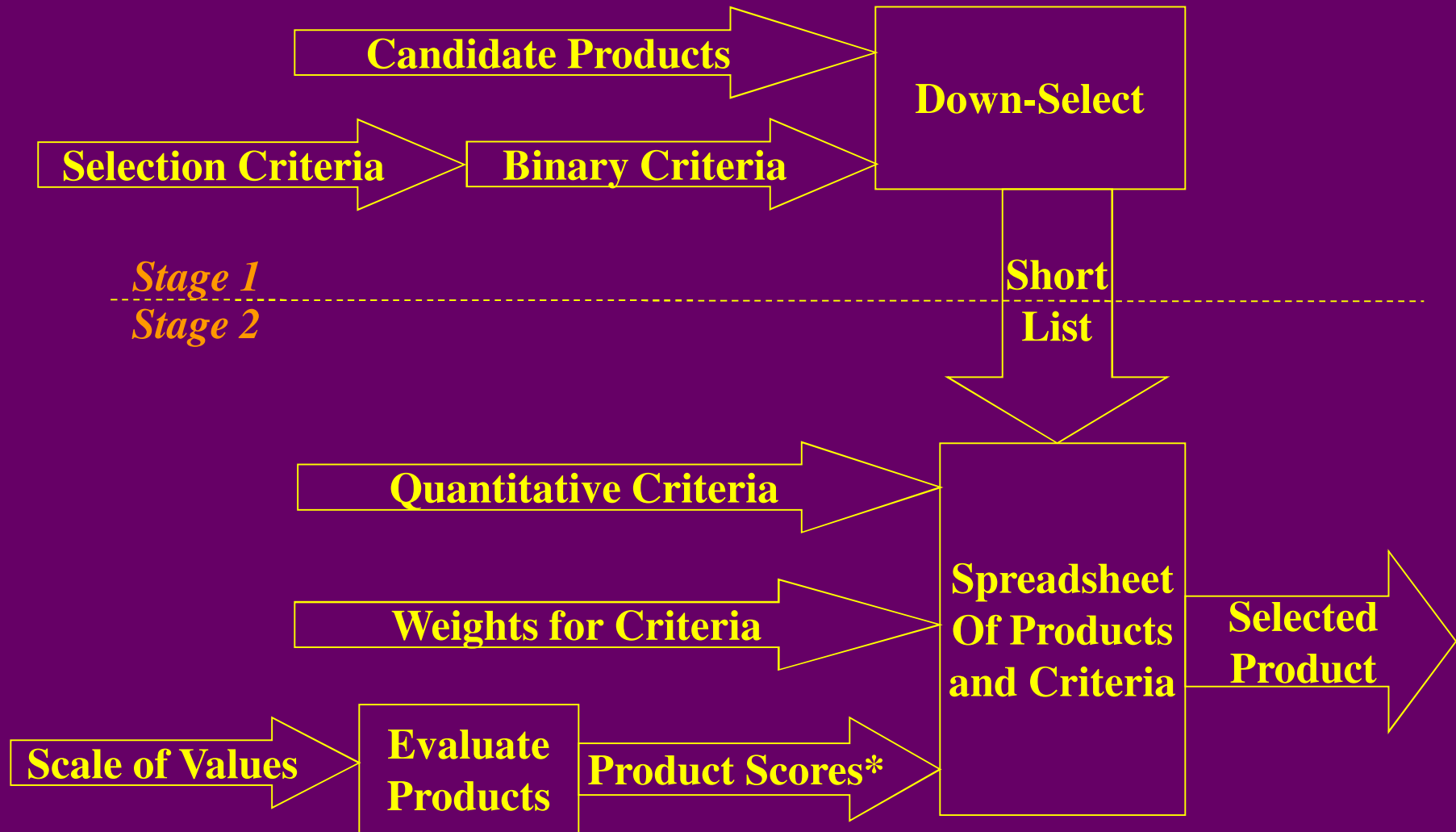
How this Process Differs

- Two stages, suitable for a large number of products
- Initial stage of binary criteria for quick down-select
- Defined numerical evaluation values
- Defined two-level weight assignment to criteria.

Two Stage Selection Process

- Identify Requirements
- Define binary criteria
- Define quantitative criteria
- Assign weights to quantitative criteria
- Identify candidate products
- Evaluate products against binary criteria – Stage 1
- Down-select using binary criteria – Stage 1
- Evaluate products against quantitative criteria – Stage 2
- Compile results to identify selected product – Stage 2.

Selection Process Stages



*** Score each product on each quantitative criterion.**

Identify Requirements

- Sources of requirements
 - Business goals
 - Domain knowledge – Existing lists of requirements
 - Stakeholders
 - Organizational and operational environment – regulations or laws that apply
- Merge and structure requirements
 - Group into logical hierarchy
 - Resolve redundancies and conflicts
 - Link each to its source(s)
 - Discuss to resolve ambiguities.

Define Binary Criteria

- Each criterion can be answered with Yes or No
 - Traceable to a requirement or otherwise justified
- These are absolute requirements
 - The product or vendor must meet the criterion
 - Any “NO” answer results in that product being rejected
 - A product with all “Yes” answers makes the short list
- Evaluate most using product literature
- Down-select prior to quantitative evaluation.

Define Quantitative Criteria

- Form criteria from requirements
 - Each criterion is an aggregate of several requirements
 - Easier to compare products to a small number of criteria
- Assign scale values that represent completeness
 - 4 = product fulfills the criterion completely or better
 - 1, 2, 3 describe parts of the criterion that are met and unmet
 - 0 = product does not address the criterion at all
- Define multiple-choice answers for each criterion
 - Each answer gets a value from 0 to 4
 - Not all values need be used for every criterion.

Quantitative Criteria Categories

- Use more than just product functions
- Some suggested criteria categories:
 - Functional
 - Human Interface
 - Performance
 - Business Continuity (explained below)
 - Life-Cycle Cost (explained below)
 - Third-Party Evaluations
 - Security.

Life Cycle Cost Category

- Product cost, relative to the other candidates
- Maintenance cost, relative to the other candidates
- Vendor's viability and ability to support the product
- Product's defect record
- Vendor's willingness to enhance or customize
- Availability and cost of training
- Compatibility with existing platforms and products

Cost in money and in time.

Business Continuity Category

- Is vendor a reputable company?
- Is the vendor company stable and likely to remain so?
- Is the product at end-of-life and about to be replaced?

Will they be around to support the product as long as we need it?

Example of Quantitative Criteria Scale Values

Seq #	The product is capable of storing and presenting for display, print, and/or download all required record information	Source or justification	0	The product is missing a provision for key record information.
			1	Most of the information and all key information can be stored, but some of the information is not easily accessed.
			2	All of the necessary information can be stored but some is not easily accessed, or some of the non-key information is not available.
			3	All of the necessary information can be stored, but some of the non-key information is not easily accessed.
			4	The product fully complies.

Functional

Example of Quantitative Criteria Scale Values

Seq #	Criteria	Source or justification	Scale Value	Description
	It is easy to print or export a full set of documentation for a specified section.		0	The information is not available in coherent form.
			2	All information is available but requires several operations to retrieve several independent pieces.
			4	The product fully complies.

Functional

Example of Quantitative Criteria Scale Values

Seq #	The product offers context-sensitive help at the screen or menu level and the field level, in addition to overall guidance	Source or justification	0	There is no Help function.
			1	Only an overall product description or guide is provided.
			2	The product includes instructional help for each screen, but it must be located within a common help file.
			3	The product offers context-sensitive help for each screen, but not at the field level.
			4	The product fully complies.

Human Interface

Example of Quantitative Criteria Scale Values

Seq #	This is the least expensive of the products being considered.	Source or justification	0	The product costs much more than its competitors
			1	Upper end of price range
			2	Middle of the price range
			3	Lower end of price range
			4	The product costs much less than its competitors

Software Life-Cycle Cost

Example of Quantitative Criteria Scale Values

Seq #	This is a solid, well-known company that has a deservedly good reputation and is doing well.	Source or justification	
			0 Company is defunct, or this is freeware with no support.
			1 Company is a start-up with no reputation, or this is open source with few interested supporters.
			2 Company is not well known, or this is open source with a network of interested supporters.
			3 Medium size company with an established reputation
			4 Blue chip company with a solid reputation and future

Business Continuity

Quantitative Criterion Weights

- Each criterion category gets a weight
 - Category weights are relative to other categories
- Criterion weights should be simple: 1 to 5
- Category weights and criterion weights can be adjusted independently of each other
- Can assign weights in parallel with product evaluations
- Preferable to have separate teams to:
 - assign weights
 - perform evaluations.

Criteria Weights—Definitions

- Category weights and criterion weights
 - Target_Weight: negotiated and assigned to a category
 - Assigned_Weight: negotiated for a criterion from 1 to 5
 - Category_Weight: calculated assuming Assigned Weights = 3
 - Final_Weight: based on Target_Weight and Assigned_Weights
- Allocate a category's weights among its criteria.
 - For example, one algorithm is:
 - $\text{category_factor} = \text{Category_Weight} / \text{criterion_count}$
 - $\text{Final_Weight} = \text{Assigned_Weight} + \text{category_factor} - 3$

Assigning Criteria Category Weights – 1

- Each organization ranks categories independently
- Conference calls discuss and negotiate rankings
- Probe into reasons behind differences in ranking
- Get each group to understand other group's reasons
- When all ranks are close, offer to split the difference
 - ranks of 3 and 5 become 4; ranks of 7 and 8 become 7.5
 - these are quasi_ranks – used to calculate Category_Weights
- Important to get buy-in from each group for the ranks.

Target_Weights assigned by consensus

Assigning Criteria Category Weights – 2

■ Set Category Target_Weights

- should total 1,000
- needs a size factor based on number of categories
- $\text{Target_Weight} = \text{size_factor} * 1,000 / \text{quasi-rank}$

■ Example with 10 categories.

- quasi_ranks: 2, 8, 10, 5.5, 2, 3, 5.5, 3, 8.5, 7.5
- $\text{size_factor} = 0.4 \rightarrow \text{Target_Weight} = 0.4 * 1,000 / \text{quasi_rank}$
- Target_Weights: 200, 50, 40, 73, 200, 133, 73, 133, 47, 53
- Category_Weights total $\approx 1,000$

Assigning Criteria Category Weights – 3

$$C_i = \frac{T_i}{\sum_{i=1}^m T_i} * 1000$$

C_i is Category_Weight calculated for that category

T_i is Target_Weight negotiated for a category

m is total number of categories

Criteria Category Weight Allocation

- The algorithm evenly allocates weights within a category
 - The middle Final_Weight values (3) for a category sum to $\text{Category_Weight} \approx \text{Target_Weight}$
 - Changing a Target_Weight recalculates all Final_Weights
 - Changing an Assigned_Weight recalculates all Final_Weights.

Criteria Category	Target Weight	Category Weight
Queries	40	40
Reporting	50	52
...
total	1002	1005

Allocate Weights Within a Category

$$F_i = \frac{A_i}{\sum_{i=1}^n A_i} * C_j$$

F_i Final_Weight of that criterion

C_j Category_Weight calculated for that category

A_i Assigned_Weight for a criterion

n is number of criteria in this category

Identify Candidate Products

- Cast a wide net—there are probably more than you know
- Products are identified in parallel with criteria definition
- Collect enough information to locate the products again.
 - Product name
 - Vendor name
 - Website URL

Evaluate Products Against Binary Criteria

- For each product, address each criterion
- Done based on product literature without installing
- Group consensus might be used for each criterion
- If any response is “No”, the product is excluded
- Products with all “Yes” answers make the short list.

Down-Select Using Binary Criteria

Criteria	Product A	Product B	Product C
Product includes the required interfaces	Yes	Yes	No
Product is available for MS Windows	No	Yes	Yes
Requires all users to be authenticated	Yes	Yes	Yes
Supports all Section 508 requirements	No	Yes	Yes
Has a web based human interface	Yes	Yes	Yes

**In this example,
only Product B goes on to the next stage.**

Evaluate Products Against Quantitative Criteria

- Evaluate each product against each criterion
- Can be done by installing and testing each product
- Can ask vendors to answer with supporting evidence
 - Evaluators validate evidence to confirm it supports the answer
- Use spreadsheets to sum product scores
 - Use a spreadsheet for each category
 - Use a totaling spreadsheet
- Product with highest score is top choice.

Evaluate Against a Criterion Category

#	Selection Criteria	Weight	Product A		Product B		Product C	
			Value	Score	Value	Score	Value	Score
1	Some criterion	7.00	2.50	17.50	4.00	28.00	4.00	28.00
2	Some criteri	6.67	3.25	21.67	0.00	0.00	0.00	0.00

	...	3.33	1.75	12.83	4.00	29.33	3.00	22.00
	Data Collection Score			559.00		809.00		775.00

Final_Weight for each criterion

weight * value = score

Category score for this product

Similar calculations are done for each category.

Evaluate Total Quantitative Criteria

Criteria	Total Score for Each Product		
	Product A	Product B	Product C
Data Collection	599.00	809.00	775.00
...
Security	453.92	657.50	589.42
...
Total Score	2086.08	2700.83	2083.75

This line totals all of the category sheets.

These numbers are links from the totals on the category sheets;

Reviewing the Hard Parts

- Define the selection criteria from the requirements
- Define scale values for each quantitative criterion
- Getting agreement on weight assignments
- Performing the product evaluations

Conclusion

- All criteria can be considered
- The decision process is documented
- The final decision is fully justified
- New information is easily added to affect the decision.