Chapter 12

Decision Making

MICHAEL J. KWINN, JR., Ph.D.
GREGORY S. PARNELL, Ph.D.
ROBERT A. DEES, M.S.

> Nothing is more difficult, and therefore more precious, than to be able to decide.
>
> —Napoleon Bonaparte

12.1 INTRODUCTION

To this point, we have learned how to work with our decision maker and our stakeholders to define our problem. We then developed candidate solutions to that problem. We now turn to the process of determining a recommendation and obtaining a decision. It is important to remember that when supporting a systems decision, which requires the level of detail of the process we describe in this book, systems engineers do not make the decision but rather they provide the necessary information to enable a logical, defensible decision by the decision maker. We will discuss how we obtain a decision later in this chapter.

In this chapter, we will first prepare to start the decision-making phase. Within this phase, we have three tasks that can be seen in the chapter's concept map shown in Figure 12.1. From the problem definition phase we have the problem statement, requirements, and value model. From the solution design phase, we have the candidate solutions and the life cycle cost model. We also may have some models and simulations that can be used in the decision making phase. The four tasks of the

Decision Making in Systems Engineering and Management, Second Edition
Edited by Gregory S. Parnell, Patrick J. Driscoll, Dale L. Henderson

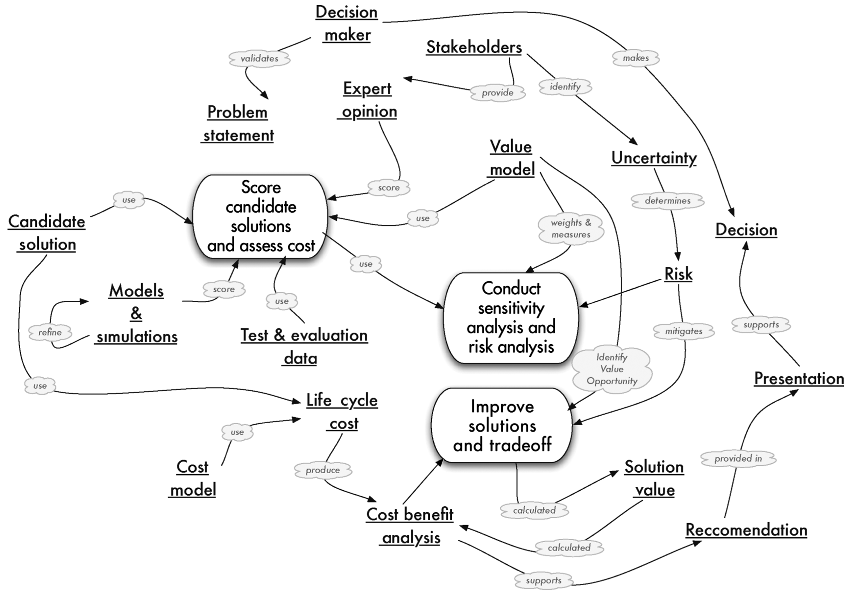


Figure 12.1 Concept map for Chapter 12.

decision-making phase are: score and cost the candidate solutions, conduct sensitivity and risk analyses, use value-focused thinking to improve solutions, and apply tradeoff analysis to compare value versus cost associated with candidate solutions. This chapter draws on material introduced by Parnell [1]. After performing these tasks, the systems engineer will present the recommended solution to the decision maker and obtain a decision. This will prepare the systems engineer for the implementation phase described in the next chapter.

Before the systems engineer begins scoring the candidate solutions, it is critical to review where we have been and ensure the process is still aligned with the problem.

12.2 PREPARING TO SCORE CANDIDATE SOLUTIONS

Like any task, before we begin the work, we have to ensure that we are prepared for success. In our case, we need to revisit some steps from previous phases of the process to ensure that we are still correctly aligned.

> Before we begin we have to review data from previous phases of the SDP to ensure the process is still aligned with the problem.

12.2.1 Revised Problem Statement

In the problem definition phase of the process, we developed a revised problem statement. Before we begin the decision-making phase of this process, we need to revisit the revised problem statement. Does the statement still capture the stakeholders' needs, wants, and desires? Does it still address the "real" problem? Is it still relevant? If the answer to any of these is no, then we need to do more work on properly structuring the problem statement. If we can still answer "yes" to each of these, we can continue.

12.2.2 Value Model

Also in the problem definition phase of this process, we developed the requirements and the value model. The requirements were used to narrow the alternatives to the potential solution candidates. The value model is very significant in the decision-making phase of the process. Though we will review the value measures and weights later in this chapter, we should pause now to review the entire value model. Is it still relevant? Does it capture all of the aspects of the problem, the functions, the requirements, and the values of the key stakeholders? If the answer to any of these is "no," we need to do some more work on the value model. If we can still answer "yes" to each of these, we can continue.

12.2.3 Candidate Solutions

We should also have our candidate solutions that form our solution set at this point in our process. The candidate solutions should have sufficient detail to allow us to score them on each value measure and assess their cost.

12.2.4 Life Cycle Cost Model

Using the principles and techniques in Chapter 5, we should have developed a life cycle cost model in the solution design phase to ensure that our candidate solutions were affordable and that they achieve any cost goals or requirements. We will use and possibly refine this model in this chapter to perform value versus cost tradeoffs.

12.2.5 Modeling and Simulation Results

When we did the solution design phase, our analysis determined the candidate solutions, and we may have accomplished some initial modeling and simulation. We should not discard these models and simulations. Some of the models and simulations we developed previously can be used in this phase to analyze candidate solutions and help improve the solutions.

12.2.6 Confirm Value Measure Ranges and Weights

Finally, this is a good time to reconfirm our ranges on the value measures and analyze their impact on the weights using the swing weight matrix for determining weights (see Chapter 10). We should have some more information from the modeling and simulation conducted to ensure the feasibility of our solutions. Do these results indicate that the ranges of our value measures may be different than we first assumed? If so, then we should make the adjustments now before we proceed to further evaluation.

So as we move forward, we have the revised problem statement, the requirements, the revised value model, the candidate solutions, a life cycle cost model, and possibly some models and simulations with associated outputs. In the steps below, we will begin to put them together to help form a recommendation for the decision maker.

12.3 FIVE SCORING METHODS

There are many methods for scoring the solutions against each value measure. Most of these fall into five categories: operations, testing, modeling, simulation, and expert opinion. We will discuss each of these, including their strengths and weaknesses.

> The five methods to score candidate solutions are operations, testing, modeling, simulation, and expert opinion.

12.3.1 Operations

The best data are usually obtained by using the system in the real operational environment. Unfortunately, operational data are usually only available for the baseline solution and candidate solutions that have been used for similar problems. It will usually be cost and schedule prohibitive to obtain operational data on all candidate solutions. For candidate solutions that require development, other scoring data will have to be used.

12.3.2 Testing

This is also known as development and operational testing.[1] This is essentially conducting testing using a prototype, a developmental, or a production system solution. Development and operational testing are key tasks in the system life cycle. We should use all available data to score our candidate solutions. Typically the development testing has less operational realism than operational testing.

In terms of accuracy and replicability, this is probably the best method to score solutions. It directly measures solutions against the value measures and reduces the number of assumptions required by the other methods.

The drawback of this method obviously is the costs associated with the scoring. Building prototypes, developmental, or production systems can be very costly and time consuming. That said, testing certainly has its important role in the design and evaluation process and is required prior to fielding any system. The other means of scoring solutions are less costly and are used in early system life cycle stages when test data are not available.

---

[1] Development testing is done by developers and operational testing is done by the users. Operational testing usually has more realistic environments. See Chapter 7 for discussion of the role of systems engineers and test engineers.

12.3.3 Modeling

Modeling usually refers to the development of mathematical models vice physical models (see Chapter 4). Mathematical models can prove to be very beneficial in the scoring of solutions. Because they are based on sound mathematical principles, they are accurate and can be easily replicated. Queuing models are important when determining service times for facilities layouts and other similar problems types. Often such problems lend themselves to a closed form solution using this type of model. If this type of problem becomes more complex, and especially if there is a stochastic nature of the problem, analysts use simulation to help determine the value measure scores. We turn to simulation now.

12.3.4 Simulation

Simulation is becoming more widely used as computing power increases and simulation packages make the building of these simulation models easier and quicker. Simulation is not limited to computer simulation, however. Simulation simply means using a representation of the solutions to determine its performance characteristics. Using this definition, simulation includes computer simulation and also physical representations of the candidate solutions (see Chapter 4).

Simulation is a very powerful tool for any systems engineer because it can be used to evaluate nearly any candidate solution. For example, a computer simulation could be built to assess the throughput of an assembly line layout. Also, a vehicle model could be manufactured similar to a candidate solution and placed in a wind tunnel or used in a physics-based virtual simulation to determine value measure scores required for the analysis.

Simulation is very useful because of its relative low cost when compared to developmental or operational testing. However, there are some significant limitations. One of the most significant is that the representation of a candidate solution may be biased by the analyst building the simulation. In other words, what is put into a simulation greatly affects what we get out of a simulation.

12.3.5 Expert Opinion

This is often considered the simplest means of obtaining the value measure scores for the candidate solutions. It is also considered the most questionable because it lends itself to more subjective analysis than the potentially more objective evaluation means described above. Certainly bias can creep into an analysis by relying too heavily on expert opinion for the value measure scores.

However, do not quickly dismiss this approach. It can be very valuable, depending on the time you have for the analysis and the level of fidelity required for the decision. For example, if the decision is required very quickly and all you have available is an expert on the subject, then this is a very sound approach to determining the value measure score. It is much quicker than building a model or a simulation and especially a prototype. It does add to the burden of sensitivity analysis, which we will discuss later in this chapter. Furthermore, experts often know operational data, development test results, and other related modeling and simulation results and use this knowledge in their assessment of the candidate solution scores.

12.3.6 Revisit Value Measures and Weights

A favorite saying of the second author is that "no value model ever survives first contact with the candidate solution scores!" After we have obtained the scores, we again return to the value measures to ensure that we can measure each of these and that the value model is weighted appropriately. Our revisiting the value measures and swing weights is critical to the success of the overall process to ensure they are correct and the weights accurately reflect the importance and impact of the variation of the value measures on the decision.

> We need to revisit the value measure ranges and update the value functions and swing weights if required.

Ensure that Each Candidate Solution Can Be Measured We begin our review of the value measures to ensure that each candidate solution can be measured against each value measure. For consistency, we would like to use the same methodology to measure each value measure against each candidate solution. This is not always possible. An example would be if a candidate solution is so early in the development stage that it cannot be fully modeled or operationally tested, but all other candidate solutions can be operationally tested.

Adjust Value Model and Weights If we have identified value measures that we cannot fully measure for each candidate solution, we have to change those value measures. This will change our value model. We should also take this time to ensure that we identify the potential ranges for each value measure. This is crucial to ensure that we have the correct weights on each value measure. In Chapter 10, we explained how to weight the value measures using the swing weight matrix. After confirming the weights, we are ready to begin our scoring of the candidate solutions.

12.4 SCORE CANDIDATE SOLUTIONS OR CANDIDATE COMPONENTS

In this section, we introduce two scoring approaches for assessing the value of competing feasible system solutions: candidate solutions and candidate components. Using the candidate solutions approach described in Section 12.4.1, we holistically score the candidate solutions for each value measure. Then we use the value functions developed in Chapter 10 to determine a value for each measure and then use Equation (10.1) to calculate the total candidate solution value. Using this approach, we calculate the value of the baseline, the candidate solutions, and the ideal solution. With a baseline, four candidate solutions, and n value measures to consider, this approach involves potentially obtaining a maximum of $5n$ scores. The baseline is a starting point for what we know. For existing systems, this could be the current system or systems used to perform the functions. In some cases, the problem is so new or technologically advanced that we many not have a baseline. The ideal solution score is the best score on each value measure. Notice that the candidate solutions approach does not evaluate all of the possible component combinations that could be made into a system.

The candidate components approach considers all feasible systems designs using the list of components to perform each system function. In this approach, we score each component using the values measures that are affected by the component. We assume that only one component contributes to each value measure's score. If we have four candidate components for each function, we would again obtain a maximum of $5n$ scores including the baseline. Next, we calculate the weighted value for the value measures affected by the component. Finally, we will use optimization (Chapter 4) to determine the candidate solution with the highest value (sum of the component value) subject to compatibility and cost constraints. The compatibility constraints may eliminate some of the component combinations that are not feasible. Section 12.4.3 describes this approach.

After obtaining scores for each candidate solution against each measure, we convert the scores to values so we can compare the solutions and develop a recommended solution decision. Some of the analysis can be done by hand, but there are software packages that can assist in more complicated analysis.

12.4.1 Software for Decision Analysis

As discussed in Chapter 1 and Chapter 9, the underlying mathematics of the SDP is multiobjective decision analysis. *OR/MS Today* has published four major surveys on decision analysis software [2]. Thirty-four of the packages trade off among multiple objectives, the focus of this chapter. Several of these packages allow the analyst to use some multiple objective decision analysis (MODA) techniques, but few can do all analyses described in this chapter. Logical Decisions™ exemplifies a typical MODA package with several built-in analysis techniques [3]. The disadvantages associated with these packages include their cost and the time needed to learn them.

Analysts can use spreadsheet models for MODA studies. Kirkwood's text offers Microsoft® Excel macros for converting scores to values [4]. Using Kirkwood's macros and spreadsheet add-ins (mathematical programming, Monte Carlo simulation, and decision trees), all the analysis described in this chapter can be performed.

Using spreadsheet models makes sense for several reasons. Spreadsheets are ubiquitous in that all computers today are configured with some office suite of software that contains a spreadsheet application. It has become the analysis environment of choice for most businesses and organizations. Consequently, clients may be more comfortable performing an analysis with spreadsheet models because they view them as less of a "black box" than more complex software. For the analyst building the model, having to construct each piece of the model builds a level of familiarity with the details and assumptions going into the model that can be useful in a decision briefing.

The main disadvantage of spreadsheet models is that analysts must design and program each model element—and this takes time. It is relatively straightforward to change a quantitative value model, such as by revising value functions, changing weights, or changing scores when the model is reasonably small. However, as a model size increases, making these updates in a spreadsheet becomes more demanding and time-consuming than similar updates using specially designed software packages. Additionally, most of the speciality software earn their value by being able to quickly produce key sensitivity analysis and summary graphics. This point will be illustrated in Section 12.6.1 for the case of using a Monte Carlo simulation for sensitivity analysis.

> There are several decision analysis software packages that simplify value models development and the analysis of candidate solutions.

12.4.2 Candidate Solution Scoring and Value Calculation

At this point, we have a value measure score for each measure for each solution. To assist in the further analysis and for proper documentation, we put this data into a table that provides scores for each of the measures for each candidate solution. This table is known as the *score matrix*, or *raw data matrix*. The raw data matrix of our rocket example is shown below in Table 12.1. For proper documentation, this table should include the dimensions for each measure and the source of the scores.

After completing this raw data matrix, we convert the raw data into the dimensionless value. This is accomplished by mapping the candidate solution's score for each measure against the value function for that measure. The resulting value is then recorded for each solution and measure in the *value matrix*. This value

TABLE 12.1 Raw Data Matrix

Candidate Solution	1.1.1 Speed of Platform (kph)	1.1.2 Percent Grade	1.2.1 Number of People	2.1.1 Thrust (lb)	2.2.1 Number of Payloads	3.1.1 Accuracy (m)	3.2.1 Range (km)
Baseline	30	20	4	1000	2	10	20
Global lightning	75	29	6	1546	5	2	100
Hot wired	66	56	3	2818	4	5	14
Star cluster	45	32	4	2993	3	4	55
Slow poke	30	42	2	1138	2	8	36
Ideal	90	60	0	3000	5	1	105

TABLE 12.2 Value Matrix

Candidate Solution	1.1.1 Speed of Platform (kph)	1.1.2 Percent Grade	1.2.1 Number of People	2.1.1 Thrust (lb)	2.2.1 Number of Payloads	3.1.1 Accuracy (m)	3.2.1 Range (km)
Baseline	14	10	40	0	10	0	19
Global lightning	77	19	5	27	100	90	94
Hot wired	65	87	60	91	60	72	13
Star cluster	35	26	40	100	20	80	52
Slow poke	13	51	90	7	10	36	33
Ideal	100	100	100	100	100	100	100

function might be discrete or continuous as described in Chapter 10. These conversions can be computed automatically using a variety of software programs such as macros in Excel or software such as Logical Decisions™. The value matrix for the rocket example is shown in Table 12.2.

The value functions convert raw data to value.

With the value matrix complete, we are ready to determine the total value for each solution.

Multiply Value by Weights and Obtain Overall Candidate Solution Values

MODA uses many mathematical equations to evaluate solutions. The simplest and most commonly used model is the additive value model introduced in Chapter 10. This model uses the following equation to calculate each candidate solution's value:

$$v(x) = \sum_{i=1}^{n} w_i v_i(x_i) \tag{12.1}$$

where $v(x)$ is the candidate solution's value, $i = 1$ to n is the number of the value measure, x_i is the candidate solution's score in the ith value measure, $v_i(x_i)$ is the single-dimensional value of the score of x_i, and w_i is the measure weight (normalized swing weight) of the ith value measure, so that all weights sum to one.

$$\sum_{i=1}^{n} w_i = 1 \tag{12.2}$$

We use the same equations to evaluate every candidate solution.

TABLE 12.3 Candidate Solution Value and Cost[a]

Candidate Solution	Cost	Candidate Solution Value
Baseline	30	10
Global lightning	85	76
Hot wired	96	62
Star cluster	48	55
Slow poke	148	26
Ideal	200	100

[a] The cost of the ideal is not known. For display purposes we assume a cost larger than the highest cost solution. An alternative would be a cost of 0.

Using the above equation, the weights for each measure developed in the swing weight matrix and the value matrix, the analyst can quickly determine the *total value* for each solution. The total value for each candidate solution in the rocket example is shown in Table 12.3 and plotted in Figure 12.5. We see that the candidate solution with the highest value is “Global lightning.” Though this solution scored the highest, we still have a great deal of work until we can make a recommendation.

Again, many software packages can accomplish this quickly by linking all the data, the weights, and the value functions. For many large-scale problems, the use of these software packages can prove to be quite beneficial.

12.4.3 Candidate Components Scoring and System Optimization

In the candidate solutions scoring approach of the previous section, we holistically scored each candidate solution on each of the n value measures. Here, we extend our analysis of the rocket problem using the candidate components approach, applying system optimization in order to determine the best system solution. As discussed earlier, in this approach we assume that only one component affects each measure. This requirement can be relaxed, but this is beyond the scope of this book. Before performing component scoring, we have to align the individual value measures uniquely to one component. Notice, however, that this many-to-one assignment allows for several value measures to be aligned with each component.

Continuing our analysis of the rocket problem, consider the component raw data matrix shown in Figure 12.2. The five component types are mobility, logistics, rocket, number of payloads, and guidance. The first type has five candidate components while each of the rest has four candidate components. Scores are provided for each measure affected by the component. To illustrate an earlier point, notice how, for example, the component “Rocket A” affects two value measures (thrust and range), but each value measure is aligned with a single component. In Figure 12.2, this means that while we might have multiple value measure row entries for a component, each value measure will be assigned to only one component.

Type of component	Components	1.1.1 Speed of platform (kph)	1.1.2 % Grade	1.2.1 Number of people	2.1.1 Thrust (pounds)	2.2.1 Number of payloads	3.1.1 Accuracy (meters)	3.2.1 Range (kilometers)
Mobility	Vehicle A	30	0.2					
	Vehicle B	75.3	0.5					
	Vehicle C	66.2	0.55					
	Vehicle D	44.7	0.32					
	Vehicle E	29.6	0.42					
Logistics	Concept A			0				
	Concept B			3				
	Concept C			4				
	Concept D			6				
Rocket	Rocket A				1000			20
	Rocket B				1500			40
	Rocket C				2000			60
	Rocket D				3000			100
Number of payloads	Two					2		
	Three					3		
	Four					4		
	Five					5		
Guidance	Inertial						7.77	
	Global positioning system						1.97	
	Wire						4.65	
	Optical						3.5	

Figure 12.2 Component raw data matrix.

Using the same value functions developed in Chapter 10 that we did for the candidate solutions approach, we next convert the raw data score on each value measure to a value as shown in the component value matrix in Figure 12.3. We then calculate each component's total value using Equation (12.1), using only the value measures affected by the component. We also calculate an ideal component value by summing the weights of the value measures affected and multiplying

		1.1.1 Speed of platform (kph)	1.1.2 % Grade	1.2.1 Number of people	2.1.1 Thrust (pounds)	2.2.1 Number of payloads	3.1.1 Accuracy (meters)	3.2.1 Range (kilometers)	Component value	Ideal
Mobility	Vehicle A	14	10						3	24
	Vehicle B	77	70						19	
	Vehicle C	65	87						16	
	Vehicle D	35	26						8	
	Vehicle E	13	51						4	
Logistics	Concept A			100					6	6
	Concept B			60					4	
	Concept C			40					2	
	Concept D			5					0	
Rocket	Rocket A				0			19	3	26
	Rocket B				25			37	8	
	Rocket C				50			56	14	
	Rocket D				100			95	25	
Number of payloads	Two					10			2	17
	Three					20			3	
	Four					60			10	
	Five					100			17	
Guidance	Inertial						36		10	27
	Global positioning system						90		24	
	Wire						72		20	
	Optical						80		22	
										100

Figure 12.3 Component value matrix.

by 100. The ideal component value is a useful analytical tool for determining the *value gap* associated with any one component. During the value improvement task, we can compare the maximum component value with the ideal component value and decide if we need to continue to search or design for higher value component candidates. For example, the highest value vehicle component (B) has a value of 19 compared to the ideal vehicle component value of 24. If this difference was considered significant, it would motivate a search for a new component that could potentially close the value gap of 5.

Once we have calculated the component value for each component, we proceed with a system optimization. The particular system optimization technique we use is adapted from the project selection methodology proposed by Kirkwood [4]. There are many useful operations research books (e.g., Ragsdale [5]) and credible websites[2] that provide detailed explanations of how to use Excel Solver. Figure 12.4 shows the standard format for the system optimization table. The table is used to calculate the value and cost of any feasible system solution. We then use Excel Solver to identify the components that provide the highest solution value in light of the design constraints imposed on the system. For the rocket problem illustrated, these constraints are budget and weight limitations.

The left side of the table is used to calculate system value. The five types of components are shown on the left and repeated on the top of the table. For the component rocket problem, we assume that we will select only one component (or one level) from each of the component categories. A binary decision variable (0 or 1) is used for each of the component categories. The resulting optimal values identified by the Excel Solver then show whether the component is selected for the best solution (1) or not (0). For example, Vehicle B is identified for use as the mobility component in the optimal system solution, whereas the rest are not.

Each cell in the system total line contains an expression representing the sum of the binary variables for each component category. These become the left-hand side of the component constraints for the Excel Solver. The required line entries are numerical values limiting this sum. For the rocket problem, these indicate that only one component can be selected for each component category. These values become the right-hand-side values for the component constraints for the Excel Solver. The value column provides the component values calculated earlier in Figure 12.3. These act as objective function coefficients for each of the binary decision variables for creating the target cell objective function for the Excel Solver. The individual component values are included in the total system value being maximized only if their corresponding component is used in the design.

The right side of the table is used to calculate the system cost and the weight of payloads and guidance that must be launched by the rocket. The five-year research and development (R&D) and production costs are included for each component. The system cost is calculated in the system total row. Again, as in the case of component value, an individual component cost is only included in the total cost if the component is used in the system. Therefore, the system total cost cells shown

---

[2] Frontline Systems, Inc. (http://www.solver.com)(http://www.solver.com) accessed August 20, 2010.

		Mobility	Logistics	Rocket	Payloads	Guidance	Value	Cost Year 1	Cost Year 2	Cost Year 3	Cost Year 4	Cost Year 5	Production cost	Weight
Mobility	Vehicle A	0					3	1	1	1	1	1	1	
	Vehicle B	1					19	2	2	2	2	2	2	
	Vehicle C	0					16	2	3	3	3	3	3	
	Vehicle D	0					8	3	4	4	4	4	4	
	Vehicle E	0					4	4	5	5	5	5	5	
Logistics	Concept A		1				6	2	2	2	2	2	0	
	Concept B		0				4	3	3	3	3	3	0	
	Concept C		0				2	4	4	4	4	4	0	
	Concept D		0				0	6	6	6	6	6	0	
Rocket	Rocket A			0			3	3	3	6	6	6	3	
	Rocket B			0			8	4	6	7	7	4	6	
	Rocket C			0			14	7	8	8	8	7	9	
	Rocket D			1			25	4	6	6	6	3	12	
Payloads	Two				0		2	2	2	2	2	3	3	4
	Three				0		3	4	4	4	4	4	6	6
	Four				1		10	4	4	4	3	2	9	8
	Five				0		17	3	3	3	3	4	12	10
Guidance	Inertial					0	10	2	4	3	3	2	10	4
	Global positioning system					1	24	1	2	1	1	1	2	1
	Wire					0	20	3	6	3	3	3	5	1
	Optical					0	22	4	3	5	6	3	15	3
	System total	1	1	1	1	1		13	16	15	14	10	25	9
	Required	1	1	1	1	1		20	18	20	20	15	50	10

System value	84
Max value with above components	91
Estimated system cost	67
Total budget	92

Legend
Decison variable: 1 or 0
Input
Calculation
Cell reference

Figure 12.4 System optimization table.

at the bottom of the table for each year contain an expression representing the sum product of the binary variables and the cost in that year. The total system production cost cell is similarly constructed. The required line for the cost columns are numerical entries listing the system annual budget limitations for R&D and production. The total payload and guidance weight is calculated in system total row. The maximum allowable weight is placed in the required row.

Four system totals are calculated in the bottom center of Figure 12.4. The system value cell contains an expression that is the sum product of the component binary variables and the component values. The "max value with above components" cell calculates the highest value that can be achieved with the component values in the table. The estimated system cost is the sum of the system total cost row on the right. The total budget is the sum of the five-year system costs and the system production cost in the required row on the right.

The system optimization using binary linear programming is performed using the following settings in Excel Solver.

- Maximize system value cell
- Decision variables: 21 component variables.
- Constraints
  - —The decision variables are binary.
  - —The number of components in the systems row on the left are set equal to the required row.
  - —The system cost row is less than or equal to the required budget row.
  - —The system payload and guidance weight is less than or equal to the maximum weigh.

Many additional constraints may arise in optimizing a system design. For example, Vehicle B is not compatible with logistics concept D, or perhaps Rocket A can only launch two payloads. Kirkwood [4] and Ragsdale [5] provide examples of additional constraints that may be appropriate for systems design constraints and many others.

Once we have determined the system value using component optimization we can plot a cost versus value plot to gain insights as to how each system solution lies in relation to each other solution, as illustrated in Figure 12.5. Since, when feasible, the optimization model will use the highest value components, the component optimization value is greater than Global Lightning. However, this high value was achieved by using more costly components.

The candidate components approach has advantages and disadvantages compared to the candidate solutions approach. The candidate solutions approach is simpler and quicker, requiring a systems engineer to develop three or four candidate solutions. Once the highest value candidate solution is identified, additional value-focused thinking is required to improve the solution. The candidate components approach requires more effort to obtain information on all of the components. Interestingly, the total number of scores generated by the two approaches is the same. The major advantage of the candidate solutions approach is that it provides assurance of identifying the highest value system solution possible given the components being considered and the constraint limitations within which all system solutions must lie.

Value-focused thinking can be used to change constraints and add better performing or less expensive components, if required by either the candidate components or candidate solutions approach. The systems engineer should select the best method to use based on availability of data and time to perform the analysis.

In the remaining sections of this chapter, we will use the candidate solutions approach. Recognize, however, that each of these sections would be similar had we chosen to use a candidate components approach.

12.5 CONDUCT SENSITIVITY ANALYSIS

When dealing with complicated decisions, such as the ones we are presenting in this book, systems engineers must be cognizant of the robustness of their analysis. The systems engineer should analyze the "sensitivity" of their candidate solution modeling assumptions and scores of the candidate solutions. The Pareto principle usually applies. Typically the decision will only be sensitive to 2 out of 10 factors. In the next section, we will discuss how the systems engineers should look at this sensitivity analysis and how it might affect their recommended solutions to a problem.

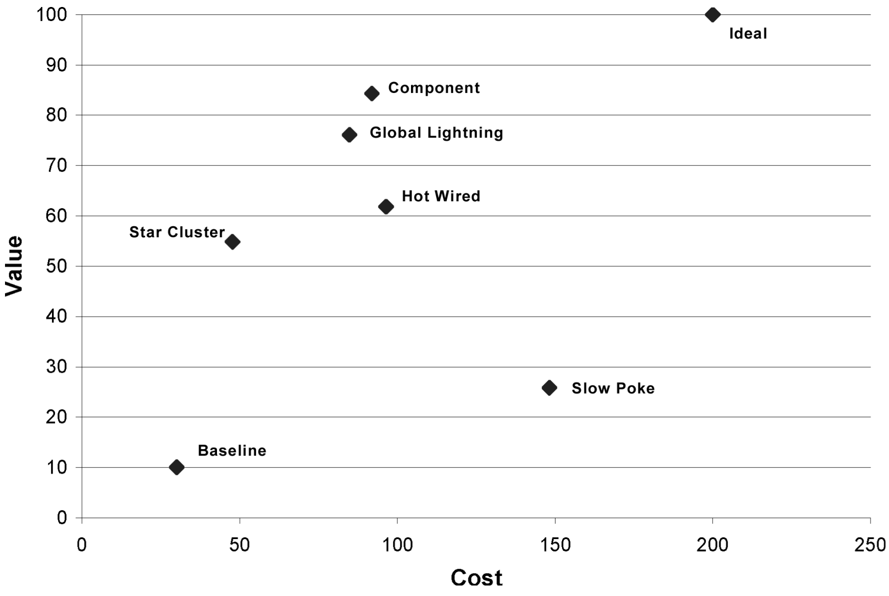


Figure 12.5 Cost versus value plot with component optimization.

> In order to tell the whole story, systems engineers must conduct sensitivity analysis to modeling assumptions and candidate system scoring uncertainty.

12.5.1 Analyzing Sensitivity on Weights

The purpose of sensitivity analysis is to see if a change in an assumption changes the preferred solution. A parameter is sensitive if the decision maker's preferred solution changes as the parameter is varied over a range of interest. The most common sensitivity analysis is sensitivity to the assessment of weights. We plot weight versus value for the range of interest for all solutions. If the solution value lines do not intersect, we say the weight is not sensitive. If the lines cross over, we consider the weights sensitive. The standard assumption for analyzing weights sensitivity is to vary one weight and hold the other weights in the same proportion so the weights still add to one. Since many factors are involved in realistically large value models (weights, value functions, and scores), typically less than 20% of the weights will be sensitive for realistic ranges of interest.

There are several ways to do weights sensitivity. The first is to vary the weight of each value measure from 0 to 1. This approach is shown in some MODA books [4] for illustrative problems with only a few value measures. This approach is not very useful for analytical purposes if there are a large number of value measures and little disagreement about the weights assessment. The second way when the weight assessments have not been controversial is to vary each of the weights by ±0.1. This is a reasonable approach to determine if small changes in the weights will change the preferred solution. However, sometimes weight assessment is controversial and key stakeholders do not want to agree on one or more weight assessments. The third way is to perform sensitivity analysis is to vary the weight across the range of interest. This is typically the most useful and important weight sensitivity analysis. If the preferred solution does not change across the range of interest, then we do not need to spend time resolving the disagreement. If the preferred solution changes across the range of interest, then we need to present this to the key stakeholders and decision maker(s) for resolution.

Using the swing weight method (see top table in Figure 12.6), there are two options for weights sensitivity analysis. The first option is to perform the sensitivity analysis using the original swing weights. This approach has the advantage that the analysis is done directly on the weight assessment judgment. The disadvantage is that the measure weight variation depends on the other swing weights. Suppose we vary a swing weight set at 85 out of 365 (= measure weight of 0.23) from 0 to 100 (the highest swing weight), the measure weight will then vary from 0 to 0.26. Applying the second approach varies the measure weight ±0.1, causing it to span the interval 0.13 to 0.33 (= swing weight assessment of 47 to 120). The advantage of this approach is that is achieves the full range of ±0.1. The disadvantage is that the swing from 100 to 120 may be unrealistic if there is agreement that the value measure with 100 has the highest swing weight.

Swing Weight Matrix with swing weights link to calculation of value:

		Level of importance of the value measure								
		Mission critical			Mission enabling			Mission enhancing		
		Value Measure	Swing Weight	Measure Weight	Value Measure	Swing Weight	Measure Weight	Value Measure	Swing Weight	Measure Weight
Range of variation	Large capability gap	Accuracy	100	0.27	Range	50	0.14			
	Significant capability gap	Speed of Launch Platform	85	0.23	Thrust of Rocket	45	0.12	Percent Grade Platform can Traverse	5	0.01
	Small capability gap	Number of Different Payloads	60	0.16	Number of Operators	20	0.05			

Total 365

Data Table with cell referencing of Solution Value calculation cells:

Sensitivity Analysis on Speed of Platform

		0	25	50	85	100
Baseline	10					
Global Lightening	76					
Hot Wired	62					
Star Cluster	56					
Slow Poke	26					
Ideal	100					

Data Table completed by referencing Speed of Platform swing weight of 85:

Sensitivity Analysis on Speed of Platform

		0	25	50	85	100
Baseline	10	9	9	10	10	10
Global Lightening	76	76	76	76	76	76
Hot Wired	62	62	62	62	62	62
Star Cluster	56	62	60	58	56	55
Slow Poke	26	30	28	27	26	25
Ideal	100	100	100	100	100	100

Figure 12.6 Performing weight sensitivity analysis.

We recommend performing weight sensitivity analysis on the value measures that have been identified as being controversial during the weight assessment across the range of interest using the swing weights. Time permitting, we also recommend varying the weight ±0.1 on the remaining measure weights.

Next we consider how to perform weights sensitivity in Microsoft® Excel.

12.5.2 Sensitivity Analysis on Weights Using Excel

Weights sensitivity can easily be performed using the table function and graphical plots in Excel. We return to the Rocket Problem. Figure 12.6 repeats the swing weight matrix from Chapter 10, Figures 10.14 and 10.15. Suppose that stakeholders had difficulty assessing the swing weight for the "speed of the launch platform." Some stakeholders thought that the weight was too high and should be much less, and some thought it could be as highly weighted as distance from target. Suppose we decide to vary the swing weight currently set at 85 out of 365 (= measure weight of 0.23) from 0 to 100 (the highest swing weight). The measure weight would then vary from 0 to 0.26.

We can perform this analysis in six steps using Excel. First, we use the weights in the swing weight matrix in Figure 12.6 to calculate solution value in a worksheet of our spreadsheet. Second, we construct the table in the middle of Figure 12.6. Across

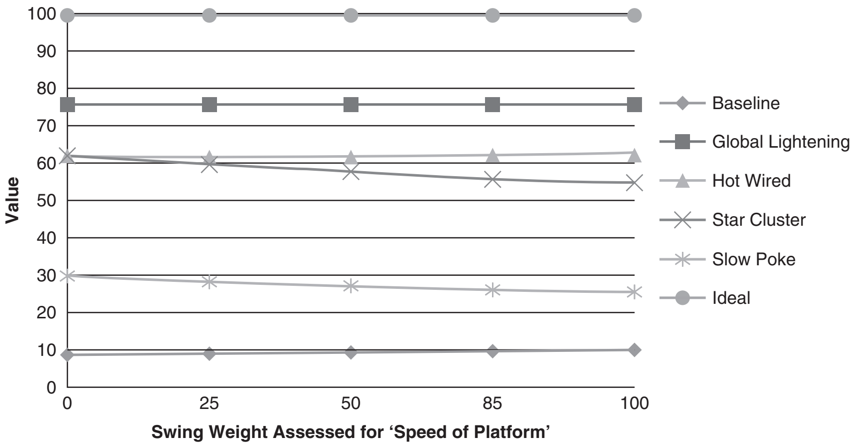


Figure 12.7 Swing weight sensitivity for Speed of Platform.

the top of the table we cell reference the swing weight name we are analyzing. In the second row, we put the swing weight range from 0 to 100 in several increments. Third, in the left-hand side of the table we cell reference the solution names and the cells used to calculate the solution value using the swing weight matrix assessments and the additive value model. Fourth, in Excel 2007 we select the "Data," "What If Analysis," and then "Data Table." Fifth, we cell reference the swing weight matrix cell with 85 in the Row Input cell in the Data Table and click on "OK." The Data Table generates the table in the bottom of Figure 12.6. Sixth, we plot the swing weight range versus solution value as shown in Figure 12.7.

Looking at Figure 12.7, we see that Global Lightning is always the highest value alternative regardless of the swing weight changing from 0 to 100. Therefore, we conclude that the preferred solution is not sensitive to the weight assessment of the speed of platform value measure. However, suppose we create a new case without Global Lightning and with an original swing weight assessment of 10 instead of 85. In this hypothetical case, the preferred solution would be Star Cluster for swing weight values below approximately 5 and Hot Wired for swing weight above 5. In this case, we would conclude that the preferred solution is sensitive to the weight assessment of the speed of platform value measure.

Swing weights are only one source of uncertainty that requires sensitivity analysis. A second source arises from our uncertainty about the solution scores on the value measures. We turn to that discussion now.

12.6 ANALYSES OF UNCERTAINTY AND RISK

In the previous section, we considered sensitivity analysis to our weight assessments. In this section we analyze the uncertainty about the scores of the candidate solutions. In the additive value model equation there are three elements: the weights, the value functions, and scores. To this point in this chapter we have assumed that the weights and value functions are known with certainty and the scores are deterministic. In many systems engineering and engineering management problems, new systems are being developed whose future performance (i.e., value measure scores) may be uncertain. In Chapter 3, risk is defined as a probabilistic event that, if it occurs, will cause unwanted change in the cost, schedule, or value return (e.g., technical performance) of an engineering system. In programmatic terms, performance uncertainty may be a major source of technical risk that could also provide schedule and cost risk.

In the rocket problem, the scores of some solutions on the value measures may be uncertain if operational or test data in operational environments is unavailable. In this chapter, probability distributions are used to assess our uncertainty about these scores and the events that can impact these scores. In the rocket problem, the impact of the uncertain scores could be a lower value than we would expect if we had used deterministic scores.

Uncertainties can impact one or more solution scores. The simplest case being considered is that each uncertainty impacts only one value measure score of one solution. We will call these *independent scoring uncertainties* and we can directly assess a distribution on each independent score. A more complex situation occurs when one uncertain event impacts the scores on the value measures of two or more solutions. We will call these *dependent scoring uncertainties* since the scores of the solutions depend on the outcome of an uncertain event.

An example of an independent uncertainty would be uncertainty concerning the accuracy of the inertial guidance used by Slow Poke (see Figure 11.5). Since Slow Poke is the only rocket using inertial guidance, this would be an independent uncertainty. In Section 12.6.1, we analyze the impact of independent uncertainties using Monte Carlo simulation. In Section 12.7.1, we analysis the impact of dependent uncertainties using decision trees [6]. An example of a dependent uncertainty would be a technical uncertainty with the effectiveness of fins. Since all solutions use fins (see Figure 11.5), a technical problem with fins could impact the range and accuracy of all the rockets.

In some analyses, we may need to consider alternative scenarios to capture our uncertainty about the future. When we consider multiple scenarios, we have some analysis options. We can consider all the scenarios and then develop one value model that captures the future planning space of the scenarios. This is the most common analysis approach. Alternatively, we could develop weights, value functions, and scores for each scenario. Clearly, this would require a lot of stakeholder time to develop multiple models and multiple scores. Since weights have a larger impact than value functions and scores, some studies have assessed different weights for each scenario and then displayed range in alternative value across all scenarios. One of the early studies using this approach was the Air Force 2025 study [7].

12.6.1 Risk Analysis—Conduct Monte Carlo Simulation on Measure Scores

When developing our raw data matrix, we identified a mean or expected score for each measure for each candidate solution. We did not identify a range of scores, but rather a single score that allowed us to calculate a total value for our candidate solutions. Oftentimes, this single score does not represent what we know—or, more precisely, *what we do not know*—of the measure.

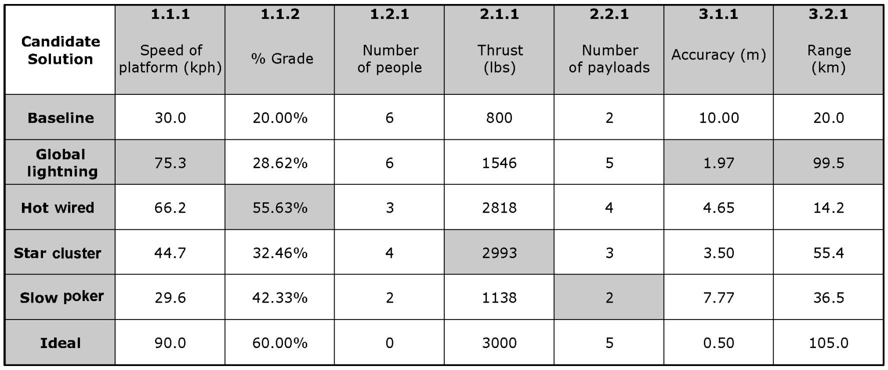
In our rocket example, we developed scores for the thrust of the rocket and the accuracy of the rocket. However, uncertainty concerning technological performance of the rocket could cause the measure scores to vary, making a reasonably accurate final score unattainable until actual testing is completed in a later life cycle stage. For our continued analysis, we could use the mean or the mode associated with these measures, but as indicators of central tendency and common occurrence, they are ineffective as indicators of uncertainty.

This uncertainty in our measure scores gets propagated into the total value of each candidate solution through the value model. If this uncertainty is significant, it can affect the profile of the cost–value tradeoff. Using a single total value for each candidate solution masks this uncertainty. An appropriate sensitivity analysis would model the uncertainty present in these measure scores enabling us to estimate the extent of total value uncertainty.

Previously, our sensitivity analysis examined the impact that changes in a single value model element (weight) had on the total value for candidate solutions. Here, our interest is in assessing the impact of simultaneous uncertainty in the measure scores so that we can examine their combined effect on the uncertainty of total value. The most appropriate tool for this is Monte Carlo simulation.

There are many software packages which can be of great assistance in this analysis. Oracle® Crystal Ball [8] and Risk™ [9] are two of the more popular modeling environments for estimating the effects of uncertainty and risk in decision making. Both of these applications are completely integrated for use with Microsoft® Excel, which has the appeal of not having to leave the environment in which the value model was created.

Illustrating this type of sensitivity analysis with Crystal Ball, we use triangular probability distributions to model the uncertainty associated with each of the six shaded measures listed in Figure 12.8. The choice of distribution to use for modeling uncertainty depends on the information elicited from key stakeholders. Without



Candidate Solution	1.1.1 Speed of platform (kph)	1.1.2 % Grade	1.2.1 Number of people	2.1.1 Thrust (lbs)	2.2.1 Number of payloads	3.1.1 Accuracy (m)	3.2.1 Range (km)
Baseline	30.0	20.00%	6	800	2	10.00	20.0
Global lightning	75.3	28.62%	6	1546	5	1.97	99.5
Hot wired	66.2	55.63%	3	2818	4	4.65	14.2
Star cluster	44.7	32.46%	4	2993	3	3.50	55.4
Slow poker	29.6	42.33%	2	1138	2	7.77	36.5
Ideal	90.0	60.00%	0	3000	5	0.50	105.0

Figure 12.8 Six measures with triangular probability distributions.

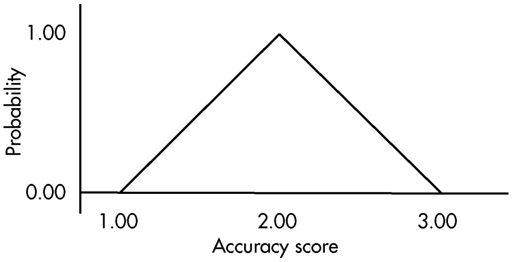


Figure 12.9 Triangular distribution for Global Lightning accuracy score.

a likeliest measure estimate, any value between some upper and lower measure score would occur with equal likelihood, motivating us to use a uniform distribution to model the uncertainty. Since this is not the case, the triangular distribution is more appropriate.

An example of the triangular distribution for Global Lightning's accuracy score is shown in Figure 12.9. The minimum accuracy of 1 meter and the maximum accuracy of 3 meters set the lower and upper limits of the distribution, respectively. The likeliest score being 2 m sets the peak location for the triangular distribution.

> The triangular distribution is commonly used to model uncertainty in Monte Carlo simulations. It requires only three parameters: a lower and upper bound to set the limits of the distribution, along with a likeliest score to establish the peak location.

For this Monte Carlo simulation we used 1000 runs to produce data with which we could estimate the variability in total value for the candidate solutions. For each run, the software samples each of the six triangular distributions to obtain a random estimate of each measure score and then calculates the total value of the six candidate solutions. When the simulation is complete, the software calculates a probability distribution on the value of each candidate solution. The probability distribution results for all six candidate solutions are shown in Figure 12.10. Two of the solutions, baseline and ideal, had no uncertain measure scores. Therefore, there is probability equal to 1.0 for each of the values. Although one of the candidate solutions, Hot Wired, had uncertainty associated with its grade measure score, the combination of a small interval between lower and upper bounds in its triangular distribution (46 to 60) and the low measure weight in the value model (0.01) impose negligible variability in its total value. Thus there is very little uncertainty being propagated into this solution's value.

The other three solutions, Slow Poke, Star Cluster, and Global Lightning, contain significant amounts of uncertainty as indicated by the variability in their total value resulting from this Monte Carlo simulation. Figure 12.10 shows a very important result: Global Lightning has the highest value but the most uncertainty. However, even with this uncertainty, Global Lightning provides a higher value than every

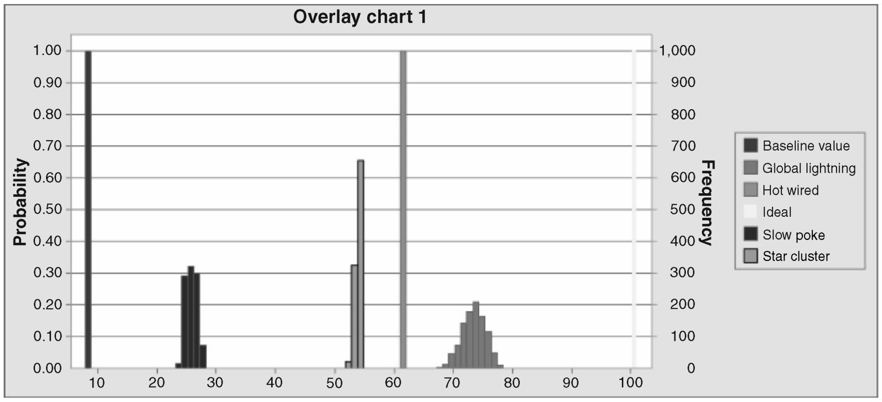


Figure 12.10 Monte Carlo simulation results.

solution except the ideal candidate solution. Its total value distribution does not overlap with any other, indicating that it *deterministically dominates* the other candidate solutions [6]. "Deterministic dominance" means that the worst outcome of the dominating solution is better than the best outcome of the dominated solution. This means that even in the face of measure uncertainty, Global Lightning can be expected to return the highest total value.

> When comparing the total value distributions from a Monte Carlo simulation sensitivity analysis, Distribution *A* deterministically dominates Distribution *B* if for every possible total value, the probability of getting a value that high is always better in *A* than in *B*.
>
> Monte Carlo simulation shows the effects of uncertainty about the scores on candidate solution values.

The next key question that the systems engineer should ask is which measure score uncertainty is making the most contribution to the variance observed in the Global Lightning value. Using Crystal Ball, we can answer this question using the Contribution to Variance Diagram shown in Figure 12.11. The measure score that creates the most uncertainty in Global Lightning value is the speed of the platform.

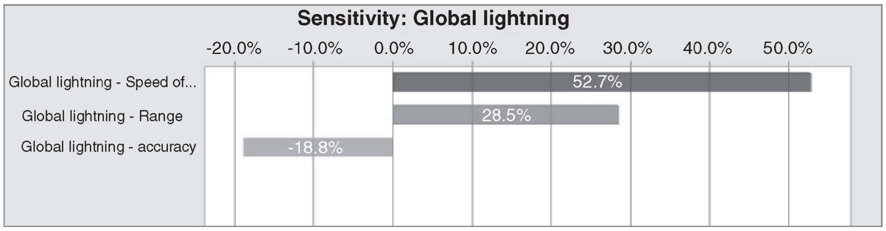


Figure 12.11 Global lightning contribution to variance.

It contributes about 53% of the variance. The systems engineer should conduct this sensitivity analysis on as many measure scores for the candidate solutions as required to ensure that the recommendation is robust for the known uncertainties. This provides the decision maker the confidence to make the system decision.

Although we have scored all of our candidate solutions and have examined sensitivity of their values to changes in weights and affects of uncertainty on the score, we are not ready to move to a recommendation yet. We first must try to develop even better solutions first using a process known as value-focused thinking [10].

12.7 USE VALUE-FOCUSED THINKING TO IMPROVE SOLUTIONS

Though we have scored all of our candidate solutions and have at least one that has scored the highest, it would be very rare indeed to have our highest value candidate solution be a perfect 100. That would mean that the candidate solution scored the highest possible for each measure. Though this is unlikely in practice, identifying an ideal solution is certainly our goal. After scoring our candidate solutions, we need to seek an even better solution.

Develop Better Solutions To seek a better candidate solution, we return to our value model. The stacked bar chart in Excel provides an excellent means of doing this. Figure 12.12 illustrates the fundamental concepts involved with applying value-focused thinking during this phase of the SDP.

In the hypothetical situation shown, the baseline is being compared with three candidate solutions. Looking at the individual measure values, we ask a couple

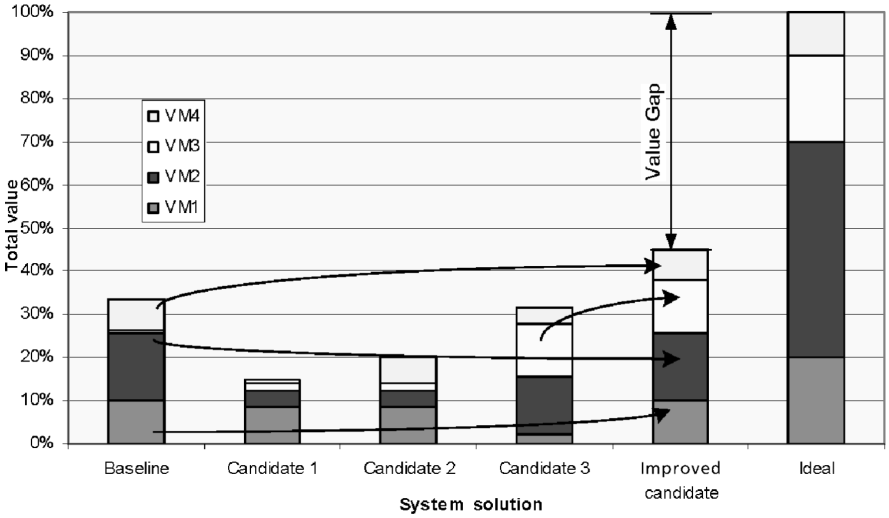


Figure 12.12 Value-focused thinking within the SDP.

of key questions. In what candidate solution do we come closest to achieving the stakeholder ideal scores? In applying value-focused thinking we first attempt to improve the best candidate solution by observing what is possible in the other candidate solutions. Assembling the maximum value measure scores into a single option produces an *improved* candidate solution. Is it possible to combine the known system elements from the different candidate solutions in this manner? What would this improved system solution look like? If it is not possible to combine the known elements generating the observed measure value levels in this manner, is there a new way of achieving a similar level of performance?

In a similar manner as described in Section 12.4.3, we notice that even this improved candidate solution falls short of the ideal levels as expressed by the stakeholders. Closing this gap will quite possibly require new design activities focusing on the candidate solution improvements that would need to be made in order to attain a better score for each measure.

It is also valuable to examine the individual measure value gaps when trying to improve candidate system solutions. Returning to the rocket design problem, Figure 12.13 shows the candidate solution stacked bar chart comparison without an improved solution. As a concept check, try assembling an improved candidate solution for this systems decision problem. By examining Figure 12.13 (and Table 12.1), we can see that to improve Global Lightning we need to improve the scores for grade, number of people, and thrust.

In order to attain an ideal score on the "% grade a platform can traverse" measure, we see that a 60% grade achieves an ideal score. What would the platform have to be to traverse a 60% grade? Can the existing platform be modified to achieve this? Does this require an entirely new platform be designed? Applying

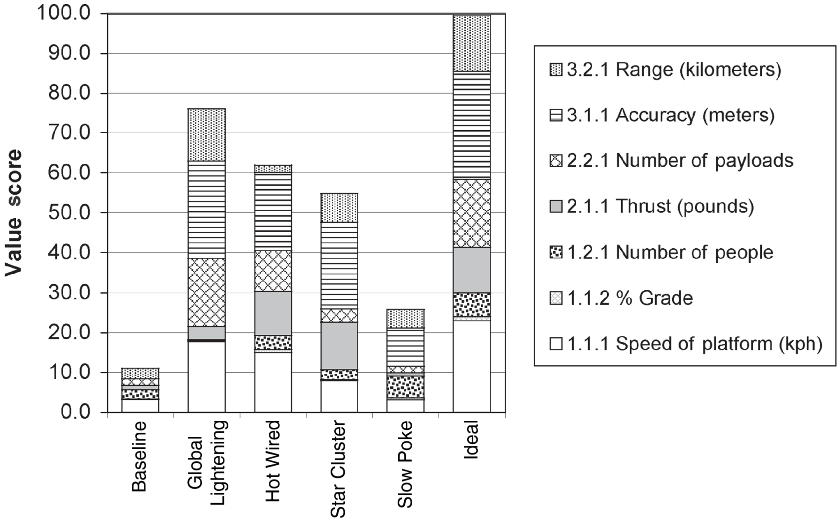


Figure 12.13 Candidate solution stacked bar chart.

value-focused thinking in this manner requires that we examine all the measures and determine what needs to be done to get closer to the ideal solution.

Once having identified the improved score for each measure, the project team works through the tradeoffs required to make any improved solution feasible. Armed with the new improved solutions, it is then time to rescore them.

Rescore the Improved Solutions When rescoring, the systems engineer can and should use scoring methodologies previously used in the process. Though we have now looked at each possible solution and analyzed the sensitivity of our process and measures, we still have to look at one other very important factor, risk, which we cannot completely eliminate. We discussed the various types of risks in Chapter 3. Here we examine how to mitigate some of the risks remaining from previous work.

12.7.1 Decision Analysis of Dependent Risks

Two major sources of uncertainty in systems development are technology development challenges and the potential actions of competitors or adversaries. Suppose for the rocket problem that two concerns are identified late in the solution design phase that could impact operational performance. The first concern is a technical concern and the second concern is a potential adversary threat.

Suppose that the engineers identify a technical concern with the new fin material that is planned to be used for both the Global Lightning and Star Cluster system solutions. This is a dependent uncertainty since the durability of the fins during flight has a direct impact on the range of the two candidate solutions. After working with the material and missile performance engineers, the systems engineer assesses the data shown in Table 12.4. If the fin material is durable, the range will achieve the original score. However, if there is some flight erosion of the fins, the range could decrease for both solutions.

Suppose that the intelligence agencies identify a potential future adversary threat that could result in degraded accuracy of guidance systems that use the Global Positioning System (GPS). Again, this is a dependent uncertainty since the accuracy of Global Lightning and Star Cluster will depend on the outcome of this event. After working with the navigation and missile performance engineers, the systems engineer assesses the data shown in Table 12.5.

We can use a decision tree to analyze the risk of dependent (and also independent) uncertainties. We use Precision Tree®, [11] a Microsoft® Excel add-in, to

TABLE 12.4 Fin Material Performance Uncertainty

		Range Score	
Fin Material	Probability	Global Lightning	Star Cluster
Durable	0.7	99.5	55.4
Erosion	0.3	80.0	40.0

TABLE 12.5 Global Positioning System (GPS) Performance Uncertainty

		Accuracy Score	
GPS Degrade	Probability	Global Lightning	Star Cluster
No Degrade	0.6	1.97	3.5
Degrade	0.4	4.0	7.0

perform the decision analysis. Decision trees are described in most decision analysis texts (e.g., reference 6), but they are typically used for single objective value and single objective utility. We use a decision tree with the multiple objective value model to determine the impact of the uncertainties on the preferences for Global Lightning and Star Cluster.

In Figure 12.14, the first node in the decision tree is a decision node, the second node is the Fin Material uncertainty, and the third node is the GPS Degrade uncertainty. Figure 12.15 shows the value calculations that are appended to the eight final branches of the decision tree. The value calculations are unchanged for the first five value measures. The value calculations for the last two value measures use the scores from Tables 12.4 and 12.5. The best decision is still Global Lightning, but the solution's expected value is now reduced from 76.3 to 71.9.

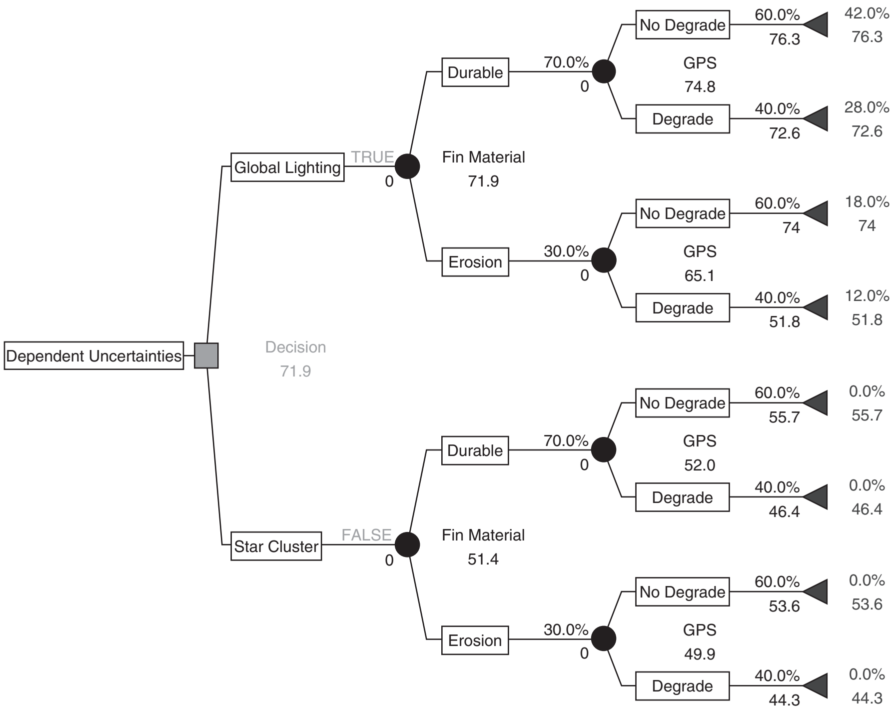


Figure 12.14 Decision tree for dependent uncertainty risk analysis.

	Speed of Launch Platform	Percent Grade Platform can Traverse	Number of Operators	Thrust of Rocket	Number of Different Payloads	Accuracy	Range	Solution Value
Global Lightning	77	19	5	27	100	90 No Degrade	96 Durable	76.3
Global Lightning	77	19	5	27	100	77 Degrade	96 Durable	72.6
Global Lightning	77	19	5	27	100	90 No Degrade	80 Erosion	74.0
Global Lightning	77	0	19	0	5	77 Degrade	80 Erosion	51.8
Star Cluster	34.6	26.2	40.0	99.7	20.0	80.0 No Degrade	55 Durable	55.7
Star Cluster	34.6	26.2	40.0	99.7	20.0	46 Degrade	55 Durable	46.4
Star Cluster	34.6	26.2	40.0	99.7	20.0	80.0 No Degrade	40 Erosion	53.6
Star Cluster	34.6	26.2	40.0	99.7	20.0	46 Degrade	40 Erosion	44.3
Weight	0.23	0.01	0.05	0.12	0.16	0.27	0.14	

Figure 12.15 Value matrix for decision tree.

Figure 12.16 shows the cumulative risk profiles for this situation. When we consider the two dependent uncertainties, we see a more complete picture of the risk of the two solutions. Since the cumulative risk profile of Global Lightning is down and to the right of Star Cluster, we conclude that Global Lightning stochastically dominates Star Cluster [6]. However, from the decision tree (and the cumulative risk profile) we see that there is a 12% probability that Global Lightning will have a value of 51.8 which is less than the original value of Star Cluster before we considered the two dependent uncertainties.

A decision maker would now be interested in knowing the impact of our assumptions about the two uncertainties. If we vary the probability that the Fin is durable from 50% to 100%, the expected value of Global Lightning ranges from 70 to 75 as shown in Figure 12.17. The decision trees, cumulative risk profiles, and sensitivity analysis are easily generated using Precision Tree®. Additional sensitivity analysis techniques—for example, two-way sensitivity—are also available in the within the software application.

This concludes our discussion of sensitivity analysis and uncertainty analysis that has focused on value. Next, we turn to cost analysis. Later we will consider value versus cost tradeoffs.

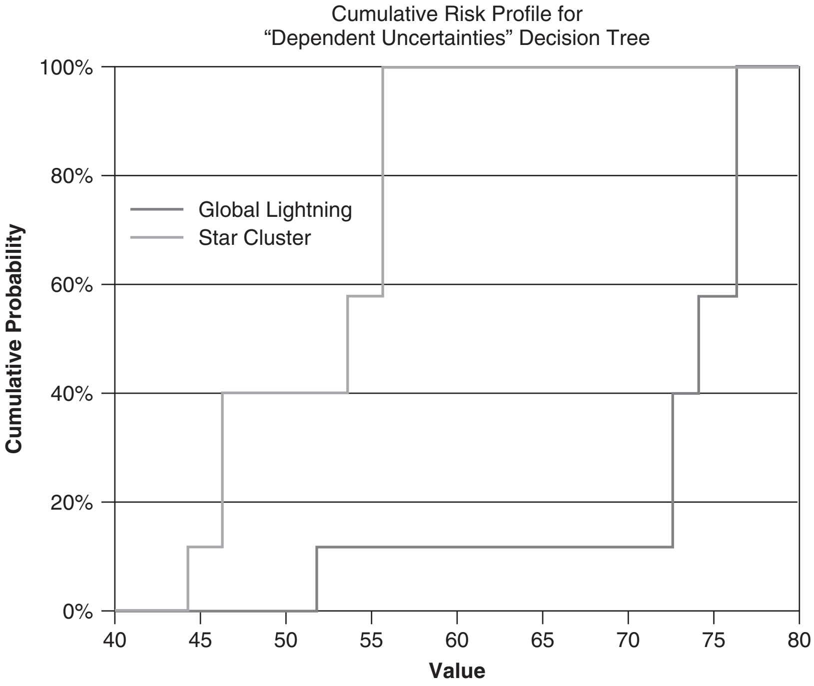


Figure 12.16 Cumulative risk profiles.

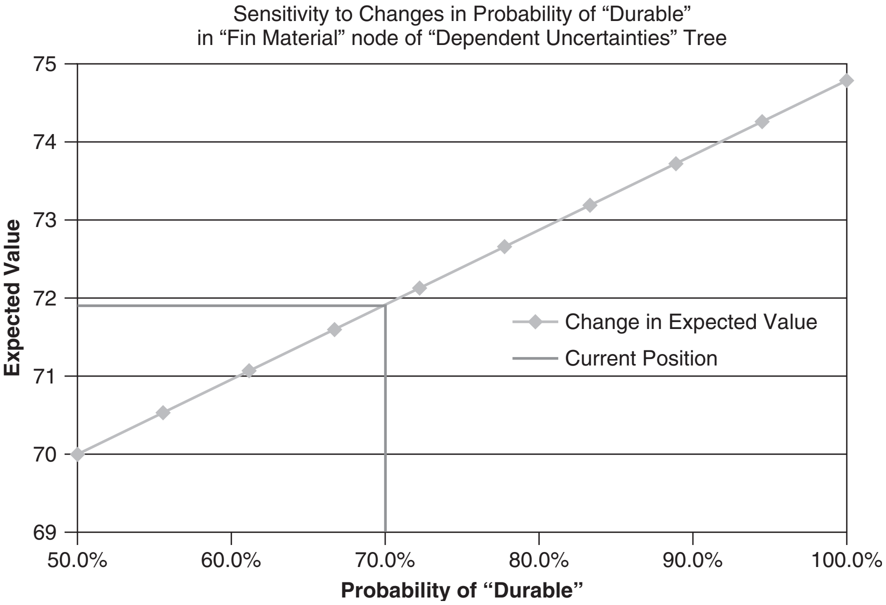


Figure 12.17 Sensitivity of dependent uncertainties.

12.8 CONDUCT COST ANALYSIS

In the Solution Design phase, after the alternatives were identified we began to develop the cost model by identifying the potential cost components and the life cycle costs using the principles and techniques described in Chapter 5. Cost analysts, system engineers, and various component engineers may have used the preliminary cost models to perform cost trades for system components or elements to improve the candidate solutions. The outputs of the Solution Design phase are the candidate solutions.

In this Decision Making phase, we continue our cost modeling and develop life cycle cost estimates for each of the candidate solutions. The systems engineer should know more about the candidate solutions at this point than when the cost components were initially developed. Typically we use two types of life cycle cost (LCC) approaches. Preparing budget estimates for a complete program typically entails preparing full LCC estimates. If detailed budget estimates are not needed, then delta LCC models are more appropriate. Delta LCC models need only estimate the solution cost deltas and not the total life cycle costs. Regardless of which approach is used, this phase is the time to draw on previous cost models and expand these models to provide additional coverage of a fuller range of costs. The key tools for cost analysts to identify the cost elements are the life cycle stages and work breakdown structures for each stage. Chapter 5 provides additional information about cost analysis tools and techniques such as production learning curve models that will be useful in later stages.

Once the systems engineer has ensured that the cost model is complete, the costs are computed for each candidate solution. Cost risks are just as important to consider as performance risks. Monte Carlo simulation can (and perhaps should) be conducted with the cost model to assess the potential cost uncertainty just as was accomplished with the value model when assessing value uncertainty. Chapter 5 describes this technique in detail.

The life cycle cost model can also be very useful in the Solution Implementation phase of the SDP. The system costs will need to be planned, executed, monitored, and controlled. The LCC model can provide a baseline for the initial plan. Cost monitoring and cost management are important implementation tasks. This will be discussed further in Chapter 13, “Solution Implementation.”

12.9 CONDUCT COST/BENEFIT ANALYSIS

Armed with the cost data, it is time to plot the cost against the benefit, or value, of each candidate solution. This approach is also known as using cost as an independent variable. It highlights to the decision maker and the stakeholders the cost versus value tradeoffs. There are instances when the candidate solution with the highest value costs significantly more than the other candidate solutions that it may not be cost effective to select that solution for implementation.

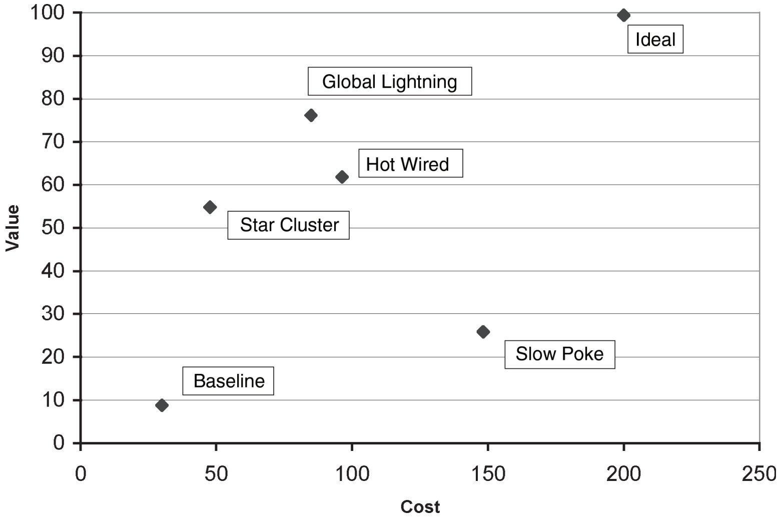


Figure 12.18 Candidate solution cost/benefit (value) plot.

Decision makers want to know the value provided for resources.

The objective is to show the decision maker the tradeoff between higher costs and higher values. This is best accomplished using a graphical representation. Figure 12.18 shows the cost/benefit chart for our rocket example. This plot is a great means to quickly convey to a decision maker the tradeoffs. From the graph, for example, the decision maker can see that he can choose the Star Cluster candidate solution which would have less value than the Global Lightning candidate solution but have a lower cost. On the other hand, the decision maker would never choose the Hot Wired or Slow Poke solutions because they have less value and higher costs than other available solutions. These two solutions are *dominated*. A dominated solution has the same value as, or a value lower than, that of another candidate solution but at greater cost than the candidate solution. Many times it is useful to put a value band (or cost band) on the chart to reflect the uncertainties.

12.10 DECISION-FOCUSED TRANSFORMATION (DFT)

When we developed the quantitative model in Chapter 10, we defined value measures based on preferences across wide, or global, ranges. This allows the systems design team to reap the benefits of identifying additional objectives during the Problem Definition phase and developing improved alternatives during the Solution Design phase [10]. As mentioned in Section 12.3.6, we should revisit the value measure ranges and associated swing weights. In practice, we often find that the candidate solutions do not span the entire global ranges specified on the value measures. The Decision-Focused Transformation [12] provides a methodology to revise value functions and swing weights prior to communication of analysis results without any additional consultation with stakeholders. When appropriate, the Decision-Focused Transformation offers the benefit of enhanced communication of analysis results by eliminating consideration of the common and unavailable value through rescaling of the value functions. Additionally, it revises our swing weights to reflect importance across these new value measure ranges. The transformation preserves the rank-ordering of the candidate solutions and the major differences of the nondominated candidate solutions. Through this, communication of results with stakeholders is enhanced, thereby increasing decision clarity.

12.10.1 Transformation Equations

The Decision-Focused Transformation is used as we prepare to present analysis to stakeholders. We have a set of alternatives, A, and their single-dimensional value, $v_i(x_{ij})$, on global value measures $i = 1, \ldots, n$. Additionally, we have measure weights, w_i, for all value measures. The magnitude of the value scale, S, may cover any range, but many use 0–1, 0–10, or 0–100. We are able to calculate $v(x_j)$, or the global value of alternative $j \in A$ using the additive value model described in Section 10.4.4. Table 12.6 provides a summary of the distinctions, descriptions, and equations used in the Decision-Focused Transformation.

In our rocket problem, we perform the necessary calculations shown in Table 12.6 to find that the common value is 7.2%, the unavailable value is 9.3%, and the discriminatory value is 83.5%. In Table 12.7, we see that discriminatory value is the sum of the discriminatory power of the value measures. This means that tradeoffs only occur within 83.5% of our original value-focused decision space. Using the last equation in Table 12.6, we are able to easily calculate the transformed total decision value of our candidate solutions. We also provide the transformation of measure weights in Table 12.7 and single-dimensional value in Table 12.8 to illustrate the transformation of the value model.

In Table 12.8, we first note that we used the equation $v'(x_j) = \sum_{i=1}^{n} w_i' v_i'(x_{ij})$ from Table 12.6 to calculate total decision value. We could also use $v'(x_j) = \frac{1}{V_d}(v(x_j) - V_c)$ without having to calculate decision measure weights or single-dimensional decision value. In the transformed single-dimensional decision value, the Hypothetical Worst is made up of the worst performances on each value measure and the Hypothetical Best is made up of the best performances on each value measure. The Hypothetical Best alternative is similar in concept to the "improved candidate" shown in Figure 12.12. These two hypothetical alternatives now bound our decision trade-space. By looking for scores of 0.0 in the transformed score data, we see that the Baseline provides equal to the worst performance on Speed of Platform, Percent Grade, Thrust, Number of Payloads, and Distance from Target. On the other hand, decision value of 100.0 indicates that the Global Lightning is the best performer on Speed of Platform, Number of Payloads, Distance from Target, and Range. When graphed in a stacked-bar chart, the differences in this data become clearer.

TABLE 12.6 Decision-Focused Transformation Distinctions, Descriptions, and Equations

Notation	Distinction	Description	Equation
r_i	Value range Utilization	Amount of the single-dimensional value scale spanned by the set of alternatives on the ith value measure.	$r_i = \max_j v_i(x_{ij}) - \min_j v_i(x_{ij})$
d_i	Discriminatory power	Weighted value range utilization on the ith value measure. Relative ability of each value measure to distinguish between alternatives.	$d_i = w_i r_i$
V_c	Common value	Sum of the weighted minimum value achieved by all alternatives on all value measures. Common to all alternatives and does not provide distinction between them; also the value of a “hypothetical worst” alternative.	$V_c = \sum_{i=1}^{n} w_i[\min_j v_i(x_{ij})]$
V_u	Unavailable value	Difference between S, the magnitude of the value scale, and the sum of the weighted maximum value achieved by all alternatives on all value measures. Not achieved by any alternative and does not provide distinction between them; summation defines value of a “hypothetical best” alternative.	$V_u = S - \sum_{i=1}^{n} w_i[\max_j v_i(x_{ij})]$
V_d	Discriminatory value	Sum of the discriminatory power of the value measures given the alternatives’ performance. Only value space where tradeoffs between alternatives occur.	$V_d = \sum_{i=1}^{n} d_i = S - V_c - V_u$
$v_i'(x_{ij})$	Decision value functions	Affine transformation of global single-dimensional value functions to a local scale defined by the alternatives in A.	$v_i'(x_{ij}) = \dfrac{v_i(x_{ij}) - \min_j v_i(x_{ij})}{\max_j v_i(x_{ij}) - \min_j v_i(x_{ij})}$
w_i'	Decision measure weights	Recalculated measure weights based on alternative scores.	$w_i' = \dfrac{d_i}{\sum_{i=1}^{n} d_i} = \dfrac{d_i}{V_d}$
$v'(x_j)$	Total Decision Value	Total value of alternative j in the local decision value model.	$v'(x_j) = \sum_{i=1}^{n} w_i' v_i'(x_{ij}) = \frac{1}{V_d}(v(x_j) - V_c)$

TABLE 12.7 Transformation to Decision Weights

DFT Element	Speed of Platform (kph)	Percent Grade (%)	Number of Operators (#)	Thrust (lb)	Number of Payloads (#)	Distance from Target (m)	Range (km)
$\max_j v_i(x_{ij})$	77.0	88.0	90.0	99.7	100.0	90.0	96.7
$\min_j v_i(x_{ij})$	14.0	1.4	5.0	0.0	10.0	0.0	13.5
r_i	63.0	86.6	85.0	99.7	90.0	90.0	83.2
w_i	0.23	0.01	0.06	0.12	0.17	0.27	0.14
d_i	14.67	1.19	4.66	12.29	14.79	24.66	11.39
w'_i	0.18	1.01	0.06	0.15	0.18	0.29	0.14
Discriminatory Value $V_d = 83.5$							

12.10.2 Visual Demonstration of Decision-Focused Transformation

With the transformation complete at the single-dimensional value function and measure weight level, we can see the effect of the decision-focused transformation on our rocket problem in Figure 12.19. In boxes 1 and 2, we display the original stacked bar chart and cost/benefit chart similar to Figures 12.13 and 12.18. In this new stacked bar chart, we also show hypothetical best/worst alternatives as defined by the best/worst performance of our candidate solutions on each single-dimensional value measure. In box 3, we display the decision-focused stacked-bar chart. The hypothetical worst now has a value of 0.0 and the hypothetical best has a value of 100.0. The transformation removed common and unavailable value. In box 4, we display the updated cost/benefit chart after the decision-focused transformation.

12.10.3 Cost/Benefit Analysis and Removal of Candidate Solutions

With the candidate solutions under consideration, we might initially conclude that our value model is relatively well-scaled in that the discriminatory value is large at 83.5% and the effect of the transformation does not drastically enhance our ability to communicate the differences between our alternatives. The decision-focused transformation is still useful when the decision is made to reduce the candidate solution set based on dominance. In our rocket problem, the Hot Wired and Slow Poke are dominated. Although an analyst does not make the decision to remove solutions, typically dominated solutions (Hot Wired and Slow Poke) are eliminated during final presentations to the decision maker. When DFT is implemented in software, we can remove these alternatives in real-time and update the transformed model. If the dominated candidate solutions are either the best or worst sole performer on any value measure, then our decision trade space will be further narrowed beyond the original transformation. As shown in Figure 12.20, we can consider the differences between the three remaining alternatives (Baseline, Star Cluster, and Global Lightning).

TABLE 12.8 Transformation to Decision Value

$v_i'(x_{ij})$	Speed of Platform (kph)	Percent Grade (%)	Number of Operators (#)	Thrust (lb)	Number of Payloads (#)	Distance from Target (m)	Range (km)	Total Decision Value ($v'(x_j)$)
Hypothetical worst	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.0**
Hypothetical best	100.0	100.0	100.0	100.0	100.0	100.0	100.0	**100.0**
Baseline	0.0	0.0	41.2	0.0	0.0	0.0	7.8	**3.4**
Global lightning	100.0	11.5	0.0	27.5	100.0	100.0	100.0	**82.6**
Hot wired	80.0	100.0	64.7	91.6	55.6	77.8	0.0	**65.2**
Star cluster	33.3	19.2	41.2	100.0	11.1	85.2	49.9	**57.0**
Slow poke	0.0	51.3	100.0	7.2	0.0	35.6	27.1	**21.5**

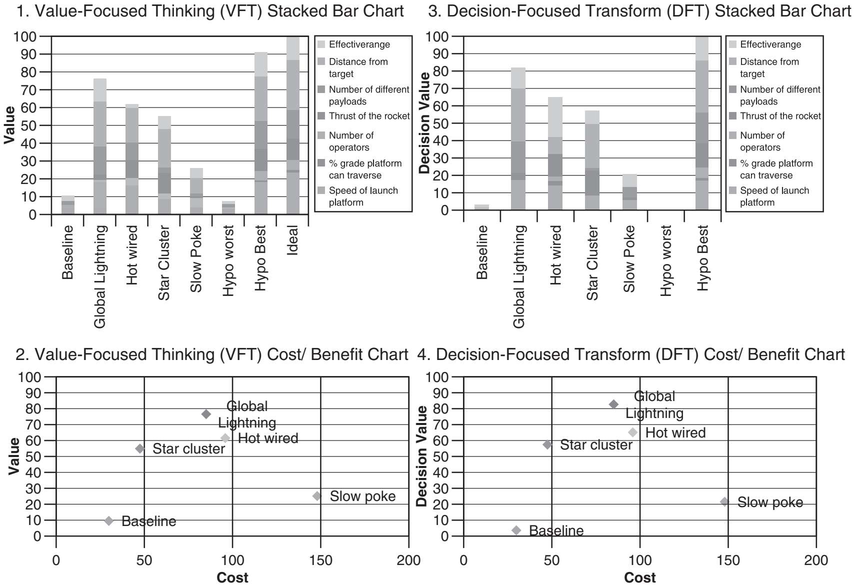


Figure 12.19 Value-focused and decision-focused results.

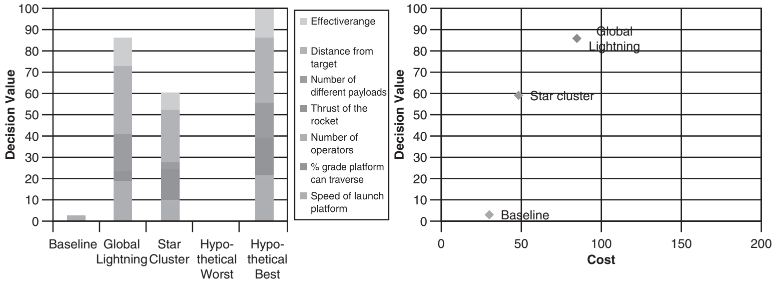


Figure 12.20 Decision-focused results after removal of dominated solutions.

With the dominated solutions (Hot Wired and Slow Poke) removed, we see that the Baseline is the worst performer on all measures other than the number of operators value measure. The absence of a bar indicates that a candidate solution is the worst performer in the remaining set. Since Star Cluster provides gains over the Baseline on several value measures for a relatively small increase in cost, it might be reasonable to believe that the decision maker would decide to remove the Baseline from consideration. If so, we can easily update the Decision-Focused model without the Baseline as shown in Figure 12.21. This second update leaves us with only two remaining solutions, the Star Cluster and Global Lighting.

Any time that we remove additional candidate solutions, we perform another distinct iteration of the Decision-Focused Transformation based on the common,

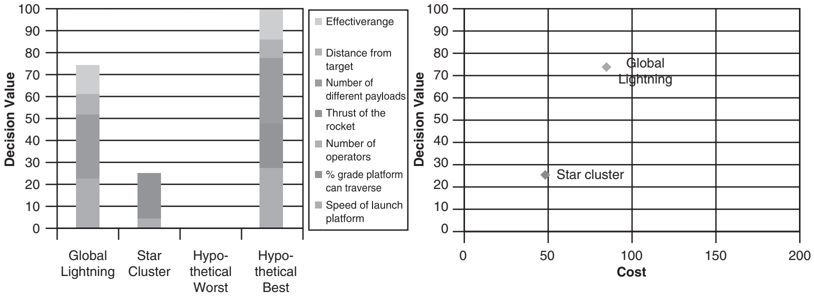


Figure 12.21 Decision-Focused results after removal of dominated and Baseline solutions.

unavailable, and discriminatory value of the reduced solution set. Now that we have only two remaining alternatives in our Decision-Focused model, we are able to see the tradeoffs clearly. Global Lightning costs more, but provides decision value on Speed of Launch Platform, Number of Different Payloads, Accuracy (Distance from Target), and Effective Range value measures. Star Cluster is less expensive and does provide decision value on Grade Platform Can Traverse, Number of Operators, and Thrust of the Rocket value measures. Although Star Cluster performs better on Grade Platform can Traverse, the difference in decision value is minor. Based on the small variation in performance between the two alternatives, the Grade Platform can Traverse measure has been reduced significantly in weight to the point where it nearly drops from our analysis; this measure does not have much discriminatory power. Discriminatory power, and the resulting decision measure weights, vividly display how measure weights depend on the importance of swinging across the established range of the value measures [12].

Additionally, we also observe that Global Lightning now has a decision value of 74.7 and Star Cluster now has a decision value of 25.3. With only two alternatives remaining, each alternative contributes to the hypothetical best for differing value measures. Their value scores now sum to 100.0, and the value scores are only based on the relative strengths between the two solutions. At a deeper theoretical level, this highlights the concept that the numeric value depends on the size of the decision trade-space. When we have a large value space initially, we see less differentiation between solutions. If we narrowly define our preferences based on a smaller set of candidate solutions, the model will display greater differentiation.

In Figure 12.21, the decision maker is presented with the simplest possible form of results and is able to consider the tradeoffs between the final two solutions with greater clarity. In the process of applying Value-Focused Thinking to build a qualitative and then quantitative model, we retain the benefits of generating additional alternatives and uncovering hidden objectives along with other benefits [1, 10]. By applying the Decision-Focused Transformation with a fixed set of solutions, we facilitate communication and clarity concerning the value and cost tradeoffs as the set is reduced, thus reinforcing commitment to action. In sum, we use Value-Focused Thinking to understand and model the complexity inherent in the situation and the Decision-Focused Transformation to simplify the understanding of tradeoffs as we move to decide.

Since Decision-Focused Transformation is a tool designed to enhance communication, we should consider the circumstances under which we would use the transformation and when it may not be useful. Dees et al. [12] prescribe that the transformation is most useful when both common and unavailable value are large (resulting in small discriminatory value) and when multiple nondominated alternatives exist. Additionally, we have shown above that the transformation is useful as we reduce the set of nondominated solutions. However, the analyst using Decision-Focused Transformation must communicate the size of the reduced decision trade-space each time the transformation is applied. Common value and unavailable value do not provide any distinction between our alternatives and cloud our ability to understand tradeoffs, but understanding their magnitudes is important. Figure 12.22 displays the magnitude of the common, unavailable, and discriminatory value after each iteration in which we removed alternatives.

As shown in Figure 12.22, we had tradeoffs in 83.5% of our value trade-space in our original Value-Focused model. When we removed the dominated alternatives, the discriminatory value reduced slightly to 79%. On the top end, we see more (12.9%) unavailable value because Hot Wired is the sole best performer on Percent Grade and Slow Poke is the sole best performer on Number of Operators.

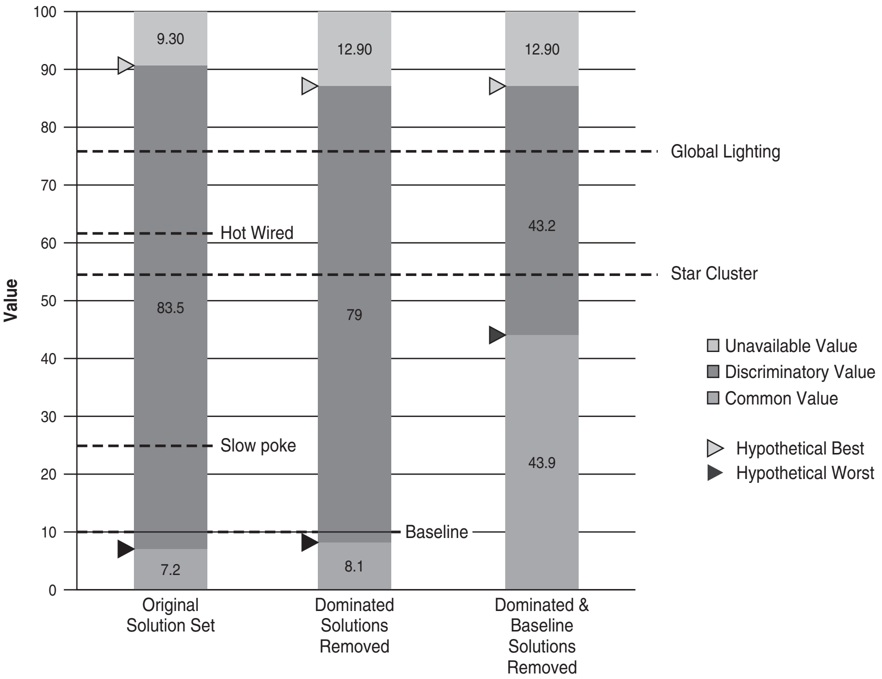


Figure 12.22 Common, Discriminatory, and Unavailable Value as alternatives are removed.

On the bottom, we see more (8.1%) common value as Hot Wired is the sole worst performer on Range and Slow Poke is the sole worst performer on Speed of Platform. In Figure 12.20, we showed that removal of the dominated alternatives along with a second iteration of the Decision-Focused Transformation produces a clearer picture of the tradeoffs than the original Value-Focused model.

Also in Figure 12.22, we show that the discriminatory value reduces to 43.2% when we further eliminate the Baseline and perform a final iteration of the Decision-Focused Transformation with only Global Lightning and Star Cluster. Unavailable value remains the same at 12.9%, but common value significantly increases to 43.9% as the Baseline is the sole worst remaining performer on Speed of Platform, Percent Grade, Thrust, Number of Payloads, Distance from Target, and Range. As we reduced the decision trade-space to 43.2% of the original size through the transformation and removal of alternatives, we must ensure that the decision maker understands we are comparing differences between solutions in a narrower band. In Figure 12.21, we offered the simplest possible tradeoffs between the final two candidate solutions in 43.2% of our original trade-space.

Low/high common value either indicates that the alternatives are all poor/good or that they are very different/similar. Similarly, low/high unavailable value indicates either that the alternatives are all good/poor or that expectations may be low/high [12]. The Decision-Focused Transformation is most useful in complex problems when there is both high common and unavailable value which imply a small discriminatory value, or decision trade-space. As discriminatory value becomes smaller, the alternatives have more similar total global value, and communicating the size of the reduced decision trade-space becomes more important.

This completes the systems engineer's analysis, but not the work. Before the systems engineer can move to the next phase in the systems decision process or the overall life cycle process, it is time to tie all the work together in a written and/or oral presentation and obtain a decision.

12.11 PREPARE RECOMMENDATION REPORT AND PRESENTATION

After all this painstaking work to develop a recommendation, it is now time to put it all together in a report or a presentation for the decision maker and stakeholders. This is a very critical step in the process. Outstanding analytical work can be quickly dismissed by a decision maker when the presentation is overly complicated or too simplistic. The perceived professionalism of a written report or oral presentation can convince a decision maker of the validity of a recommendation, confuse the decision maker into inaction, or motivate them to find a better systems engineer.

There is no one set order for developing the report or making a presentation. Though some decision makers request a decision briefing and then want a follow-up report that includes the decision and implementation plan, most decision makers request a final decision briefing accompanied by a detailed report. The implementation plan is then developed separately. We assume that this latter case is addressing what follows. While we do make several specific suggestions concerning the format of a presentation to increase its usefulness to the decision maker, our primary objective here is to provide general guidance on important factors that give a report and presentation the greatest chance of success.

12.11.1 Develop Report

Organizations will often establish a standard format for written technical reports. Standardizing the content and format lessens the burden on an analyst who has to prepare the report and makes it easier for decision makers to locate specific items of interest in the report. Regardless of the existence or absence of a specific format, there are some basic principles in the development of a technical report for a decision maker. Reports should include an executive summary, main body, and appendices (as necessary). These are in order in the report and in the order of detail.

The key to a successful technical report is a clear, concise executive summary.

Executive Summary The executive summary is designed to provide a brief overview of the content of the report. It provides the decision maker with enough supporting facts to make a decision without having to read the entire report. It should include the objective of the report (often to obtain a decision), the most compelling evidence to support a decision, and a quick overview of the methodology used. The best executive summaries can be crafted to fit on a single page. It should very rarely be longer than 10% of the overall length of the main body or over five pages, whichever is less. Additional details are provided in the body of the technical report.

Main Body The main body of the report is designed to be a much more detailed explanation of the study. Here the systems engineer must tell the story of what the analysis means to the decision maker and key stakeholders. This is a technical report. The writing should be very concise and restricted to the important parts of the analysis which support the recommendation. It should be organized to allow the decision maker and key stakeholders to follow the analysis from the initial problem statement until the recommended decision. Rarely should the systems engineer include steps not taken. For example, if the analysis did not lend itself to operational testing, the analyst should not include a paragraph on operational testing even if the organization's standard report format calls for such a paragraph. The only exception would be if the absence of this step has a significant impact on the recommendation (e.g., in risk mitigation).

The main body should be detailed enough for understanding the analysis conducted and how it supports the decision, but should refrain from being so detailed that the analysis obscures the recommendation.

Appendices The appendices of the report should include detailed formulations of models, simulation code, and data. These are rarely of interest to the decision maker, unless he or she is extremely technical or there are questions in the analysis, but are very useful to other analysts or stakeholders. A decision maker may ask other analysts to comment on your report, in which case, these appendices are very important.

The final crucial part of any written work is the documentation and references to any support received or researched in the analysis. Proper documentation provides two things. First, it provides credibility to the work as it shows support by previous respected work. Second, and most important, it supports the integrity of the analyst. Nothing destroys an analysis as quickly as questionable documentation and even experienced systems engineers cannot easily recover from integrity problems.

12.11.2 Develop Presentation

The single most important consideration when developing a presentation is understanding what the decision maker needs in the presentation in order to make a decision. Written reports are commonly tailored to the type of problem being addressed and accepted report format of an organization. Oral presentations must be tailored to the decision maker. They must include the detail required to make a decision and to capture and hold the interest of the decision maker throughout the presentation. There are some general guidelines to follow when constructing a presentation used to obtain a decision.

> The most successful presentations stay on message and stay within time limits.

Opening The opening should set the tone for the remainder of the presentation. The presenter should immediately state the purpose of the presentation to focus the expectations of the decision maker. In this case, the purpose is to obtain a decision. Immediately following the purpose, the presenter should provide the decision maker with enough background on the problem to frame and focus their attention on the topic being presented. Although the topic may be fresh in the mind of the presenter, the decision maker may have just left a situation involving a topic entirely different than the one at hand. The presenter should explain why the problem and the current presentation are important to the decision maker.

The final part of the opening is the recommendation. This is known as “the Bottom-Line Up Front” (BLUF). This provides the decision maker with both a good idea of where the presentation is heading and the recommended solution decision. Knowing the final recommendation helps the decision maker focus on the questions critical to the decision that he or she will make.

Presentation of Analysis The presenter should start the description of the analysis from an accepted point of common knowledge. This might be a summary of the previous meeting or even going back to the original problem statement. This allows the decision maker to feel knowledgeable and comfortable at the start of the discussion of the analysis.

From there, the briefing should take the decision maker through the process at a detail required to maintain his or her interest and understanding until the recommendation is reached. Some decision makers are very detail-oriented and want the formulations and the data. Some want only highlights. In the absence of prior knowledge, the presenter should present limited details and have backup information ready to address specific questions.

The presentation should logically flow from the start point until an ultimate conclusion. This keeps the decision maker knowledgeable and comfortable. A knowledgeable and comfortable decision maker will be much more likely to support an analysis and make a decision at the end of a presentation than a decision maker who is overwhelmed with information and confused. This decision maker is more likely to put off a decision rather than make a wrong decision.

A good presenter will know exactly how much time the decision maker has available and will keep the briefing shorter than the time allotted. This allows more time for questions, and a busy decision maker will appreciate the extra time. Do not assume that the decision maker will allocate extra time for this presentation because he or she may leave prior to making a decision. Always have a one chart summary if the decision maker has to shorten the time.

> Always have a one chart summary in case the decision maker has to shorten the time.

Concluding the Presentation After presenting a concise and convincing argument, the presenter should restate the recommendation and ask for a decision. When a presenter states at the start of the presentation that the purpose is to obtain a decision, the decision maker will be prepared for this request. The decision maker might want to put off the decision; and, if so, the presenter should politely ask when a decision might be forthcoming. Though some decision makers do not like to be pressed, when the timing of the decision is critical (e.g., in the progress of a manufacturing or development process), it is worth the effort to press the issue.

Whether the decision maker makes a decision or not, the presenter should continue with the future actions required based on the decision or lack thereof. Since the decision may significantly change the information prepared in advance of the presentation, the presenter should be prepared to adjust the plans as necessary.

Some final thoughts on briefings:

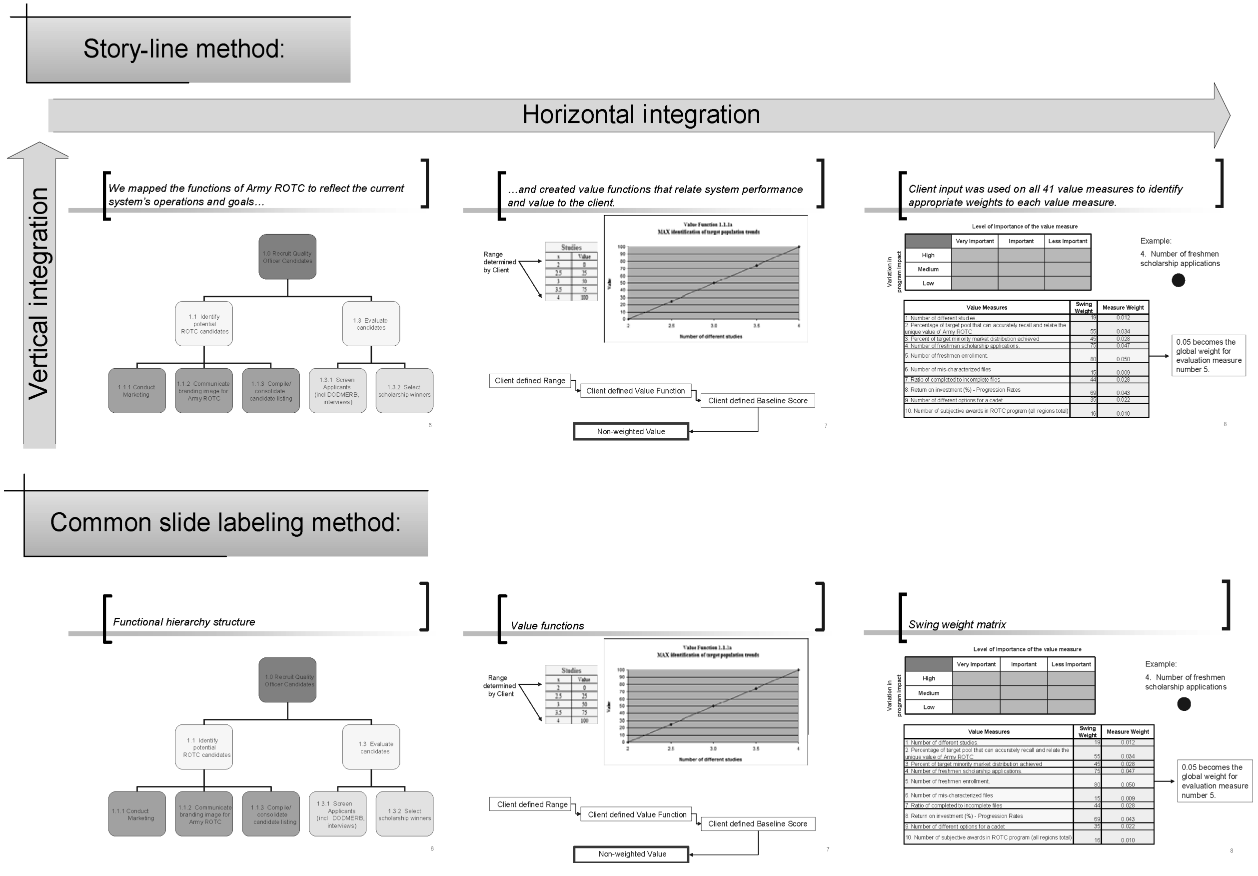
- *Do Not Read the Slides*. Nothing detracts from a presentation and infuriates an audience as quickly. Summarize the slide or the chart.
- *Have Simple Slides and Quick Thoughts*. A slide or concept in a presentation that tries to convey too much information often loses the audience and conveys little.
- *Transition the Decision Maker to Focus on the Problem Topic Very Early in the Presentation*. Yours is not the only problem on their mind.
- *Be Careful with the Use of Pointers*. These can often distract the audience from the presentation, especially if the presenter is nervous!

- *Keep Text Font Size Consistent Throughout the Presentation*. Using larger font sizes has an effect similar to that of capitalizing letters in e-mail: giving the impression of yelling.
- *Dress Professionally*. The presenter should always be more formal than the decision maker, but not overly so.
- *Speak Professionally*. Do not use quaint or colloquial phrases or try to be too funny. The briefing is designed to obtain a decision, not audition for stand-up.
- *Stay on Message*. Do not introduce tangential material that is not essential to the decision.
- *End in Control of the Presentation in a Way That Lets the Decision Maker Know Where the Project Is at, Where It Is Going, When Deliverables Should Be Expected, and What Actions Are Required of the Decision Maker in Order to Make the Project a Success*. It is a parting shot to reframe the presentation content before you lose the decision maker's attention.

Using a Storyline Approach One straightforward method for organizing information and presenting it effectively using is called the *storyline* method. There are two principles invoked when creating a presentation using this approach: horizontal integration of the story and vertical integration of support. Conceptualizing each presentation slide as a single page of a book, the area at the top of the slide typically reserved for a slide title is used to "tell the story" of the presentation content from start to finish using a single sentence on each slide. Done correctly, the decision maker should be able to read across the top of every slide and understand the main messages that the system team wants to convey. This effect is known as achieving *horizontal integration* in the sense that if all the slides were laid out on a table in order, the presentation storyline could be read by reading horizontally across the slides.

The main body of each slide is then used to present key evidence (e.g., text, graphics, mathematics, simulation results, etc.) supporting the storyline sentence present in the slide title area. This is known as achieving *vertical integration* of the presentation material. It is "vertical" in the sense that the typical audience member will logically look to the title area first, encountering the storyline statement, and then "drill down" into the supporting evidence below the statement to understand the logical basis for the statement. Figure 12.23 illustrates a comparison between the storyline approach and a typical default presentation format that uses simple labels as slide titles in Microsoft® PowerPoint.

One attractive feature of this method is that it forces a presenter to clearly address the salient points needing to be made, the logic connecting these points, and the key elements of convincing evidence that the statement is factually based in its claim. This frees the presenter to add value during the presentation by providing the audience with insights and reasoning that complement what they are seeing instead of reading the content of the slides to the audience, which is considered bad practice.



We mapped the functions of Army ROTC to reflect the current system's operations and goals…

…and created value functions that relate system performance and value to the client.

Studies	
x	Value
2	0
2.5	25
3	50
3.5	75
4	100

Client input was used on all 41 value measures to identify appropriate weights to each value measure.

Level of Importance of the value measure

	Very Important	Important	Less Important
High			
Medium			
Low			

Example:

4. Number of freshmen scholarship applications

Value Measures	Swing Weight	Measure Weight
1. Number of different studies.	19	0.012
2. Percentage of target pool that can accurately recall and relate the unique value of Army ROTC	55	0.034
3. Percent of target minority market distribution achieved	45	0.028
4. Number of freshmen scholarship applications.	75	0.047
5. Number of freshmen enrollment.	80	0.050
6. Number of mis-characterized files	15	0.009
7. Ratio of completed to incomplete files	44	0.028
8. Return on investment (%) - Progression Rates	69	0.043
9. Number of different options for a cadet	35	0.022
10. Number of subjective awards in ROTC program (all regions total)	16	0.010

0.05 becomes the global weight for evaluation measure number 5.

Functional hierarchy structure

Value functions

Swing weight matrix

Studies	
x	Value
2	0
2.5	25
3	50
3.5	75
4	100

Level of Importance of the value measure

	Very Important	Important	Less Important
High			
Medium			
Low			

Example:

4. Number of freshmen scholarship applications

Value Measures	Swing Weight	Measure Weight
1. Number of different studies.	19	0.012
2. Percentage of target pool that can accurately recall and relate the unique value of Army ROTC	55	0.034
3. Percent of target minority market distribution achieved	45	0.028
4. Number of freshmen scholarship applications.	75	0.047
5. Number of freshmen enrollment.	80	0.050
6. Number of mis-characterized files	15	0.009
7. Ratio of completed to incomplete files	44	0.028
8. Return on investment (%) - Progression Rates	69	0.043
9. Number of different options for a cadet	35	0.022
10. Number of subjective awards in ROTC program (all regions total)	16	0.010

0.05 becomes the global weight for evaluation measure number 5.

Figure 12.23 Two methods for organizing and presenting information in slideshows [13].

The storyline method delivers two additional benefits that add to its appeal for presentations supporting systems decision making. First, it is not uncommon for decision briefings to be circulated widely throughout an organization after the presentation concludes. Vertical and horizontal integration helps prevent individuals who were not present for the presentation from misinterpreting the message because the main points are clearly present along with their supporting evidence. Similarly, the storyline method enables slide handouts to function as stand-alone references for the presentation at a later date.

Secondly, slides created using this method tell the intended story. The resulting presentation can serve as a logical template for creating a technical report on the topic as well. Using the presentation in this fashion requires each slide to be first placed on a single page in a document. Next, one elaborates on the title line statement and describes to the reader the evidence providing vertical integration support to the statement. Any graphics or images required to make the important points clear are retained as figures and charts in the technical report. Any slide images that contain purely textual information will eventually be deleted, replaced by the expanded description crafted in the report body. Adding any necessary references and section organization nearly completes the report.

Lastly, a storyline approach is very helpful to “story board” the flow of the presentation prior to implementing it in software. One way of doing this when classroom or conference facilities are available is as follows. Estimating as a rule of thumb that every slide will consume approximately a minute of presentation time on average, draw an empty box (placeholder slide) for each minute of the presentation. Below each slide, block out and identify groups of slides that will contain the general content of the presentation as the team intends it to unfold. These contents consist of, but are not limited to: title slide/team identification, agenda, current project timeline, bottom-line up-front (optional but encouraged), problem background and description, methodology, modeling, results and analysis, conclusions, recommendations, areas for further improvement, updated timeline, references (optional). The logical organization of these placeholder slides aligns with the horizontal integration of the slides when the presentation is complete.

Next, identify the content of each slide (in general terms, not specific detail) needed to support the storyline. The idea here is to see the presentation from a single, macroscopic perspective in the hope that by doing so any gaps in logic, analysis, content, and so on, will be revealed. Finally, by examining the information the team actually possesses to support the storyline, the team’s workflow can be adjusted as necessary to fill-in any missing information prior to the presentation being given.

Presentation software is not the only choice for conducting effective presentations. Very successful briefings can be conducted using butcher charts, simple paper slides, or even chalk. The key is that the presentation is professional and it is concise. A decision maker will appreciate a presentation much more if it conveys a simple message than if the words come flying in from the side and there are explosions and movies. Many experts suggest that slides or charts should include no more than three ideas and have fewer than four very short lines per slide, chart, or board space.

12.12 PREPARE FOR SOLUTION IMPLEMENTATION

In this chapter, we have worked through the decision making phase of the system decision process. We started with the important items developed in the previous phases of the process, the revised problem statement, the requirements, the value model, some cost models, the candidate solutions and some previous results of the initial analysis from the modeling and simulation and testing efforts. After reviewing these important elements of our analysis to ensure they are still relevant, we were ready to proceed.

We identified the data we were going to use to complete the raw data matrix. We then used the value model to convert the raw data to the values and obtained the value for each candidate solution. After evaluating the risks, the sensitivity of our analysis and the costs of our candidate solutions, and developing improved solutions, we were ready to develop a recommendation.

After developing a recommendation, we prepared written reports and oral presentations in order to obtain a decision from the decision maker. We now have to determine what we are going to do with the solution decision. Depending on where in the system life cycle the process has been employed, the systems engineer must now develop a plan to implement the solution decision. This leads to the next and final step in the systems decision process: solution implementation. The systems engineer will find that the easy part of the process has been done. The implementation is the difficult part of process.

12.13 ILLUSTRATIVE EXAMPLE: SYSTEMS ENGINEERING CURRICULUM MANAGEMENT SYSTEM (CMS)—DECISION MAKING

Robert Kewley, Ph.D. U.S. Military Academy

Decision Making

Once they had alternative solutions from the solution design phase of the system concept decision, the design team had to score those solutions against the values developed during the problem definition phase in order to come up with a recommended decision to present to the department leadership, the customer, for funding. In order to score each candidate solution, the design team used research from the solution design phase to evaluate each objective using the measures of effectiveness. The raw data matrix in Figure 12.24 shows the results of this subjective assessment.

The constructive scores had the following values:

−1 Worse than current system
0 Same as current system

			Solutions				
	Objectives	Measures of effectiveness	Improve existing system	Contractor development	Web data development	Content management (A)	Content management (B)
Value matrix	Maximize content sharing	Constructive scale comparison to current system	1	2	2	3	3
	Enforce standards	Constructive scale comparison to current system	1	2	2	2	2
	Maximize availability	Constructive scale comparison to current system	0	0	0	-1	-1
	Maximize collaboration	Constructive scale comparison to current system	1	2	1	3	3
	Maximize usability	Constructive scale comparison to current system	1	1	1	3	2
	Maximize flexibility	Constructive scale comparison to current system	-1	0	1	3	2
	Maximize security	Constructive scale comparison to current system	0	2	2	2	2
	Minimize dev. and maint. time	Months to achieve course-level functionality	1	6	4	2	3
	Minimize dev. cost	Total development cost in dollars	$0	$250,000	$1,000	$2,000	$0
	Integrate	Constructive scale comparison to current system	0	1	1	2	2

Figure 12.24 Raw data matrix for CMS system concept decision.

+**1** Marginal improvement to current system
+**2** Some improvement to current system
+**3** Significant improvement to current system

In order to determine the total value of each candidate solution, these raw data scores had to be converted to value scores (on a scale of 0 to 10) using the value functions developed in the value modeling step of the problem definition phase. The results of this transformation are shown in Figure 12.25.

In order to get a total value for each alternative, the design team calculated the value of each candidate solution using the additive value model [Equation (9.1)]. Figure 12.26 shows a graph of these results.

This analysis shows that the content management system for Vendor A is the best solution for providing IT support to the curriculum management functions in the Department of Systems Engineering. A closer look at the measures of effectiveness shows that the usability and collaboration capabilities of content management systems give them significantly more value than other forms of development. This is primarily due to built-in capabilities for file management, discussion, survey response, and e-mail. Furthermore, Vendor A provides an advantage over Vendor B with respect to instructor ease of use, flexibility, and integration. The design team assessed Vendor A's product to provide more drag-and-drop functionality, more user customization features, shorter development time to tweak or add features, and a better capability to integrate with other Academy-wide IT systems.

			Solutions				
	Objectives	**Measures of effectiveness**	Improve existing system	Contractor development	Web data development	Content management (A)	Content management (B)
Raw Data Matrix	Maximize content sharing	Constructive scale comparison to current system	3	6	6	10	10
	Enforce standards	Constructive scale comparison to current system	3	6	6	6	6
	Maximize availability	Constructive scale comparison to current system	4	4	4	0	0
	Maximize collaboration	Constructive scale comparison to current system	3	6	3	10	10
	Maximize usability	Constructive scale comparison to current system	5	5	5	10	7.5
	Maximize flexibility	Constructive scale comparison to current system	0	2.5	5	10	7.5
	Maximize security	Constructive scale comparison to current system	4	9	9	9	9
	Minimize dev and maint time	Months to achieve course-level functionality	8.3	0.0	3.3	6.7	5.0
	Minimize dev cost	Total development cost in dollars	10	0	10	10	10
	Integrate	Constructive scale comparison to current system	1	3	3	6	6

Figure 12.25 Value matrix for CMS system concept decision.

Because usability is such a significant factor in the total value, the design team performed sensitivity analysis to see how the scores might change if this factor were not weighted so heavily. This analysis calculated the resulting total value if the importance of usability in swing weighting were medium or low, instead of high. The resulting swing weighting values would be 75 and 20, instead of 100. Figure 12.27 shows the results of this analysis. The ordering of the alternatives did not change as collaboration importance ranged from very important to important to less important. This gives increased confidence in the recommendation. Similar analysis was done for other measures of effectiveness.

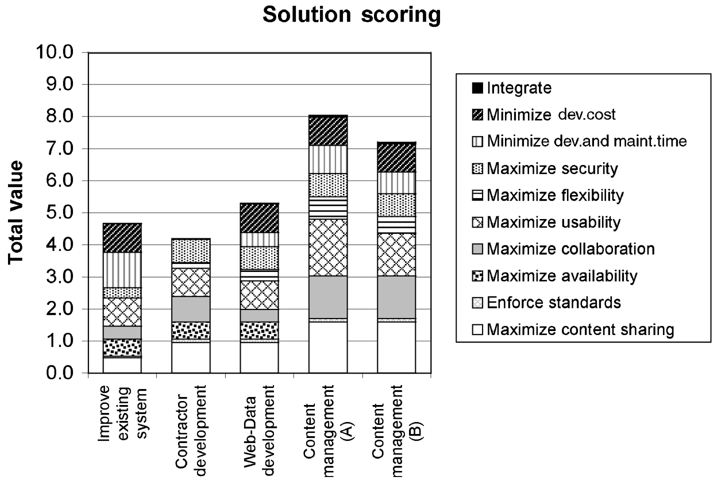


Figure 12.26 Solution scoring for CMS system concept decision.

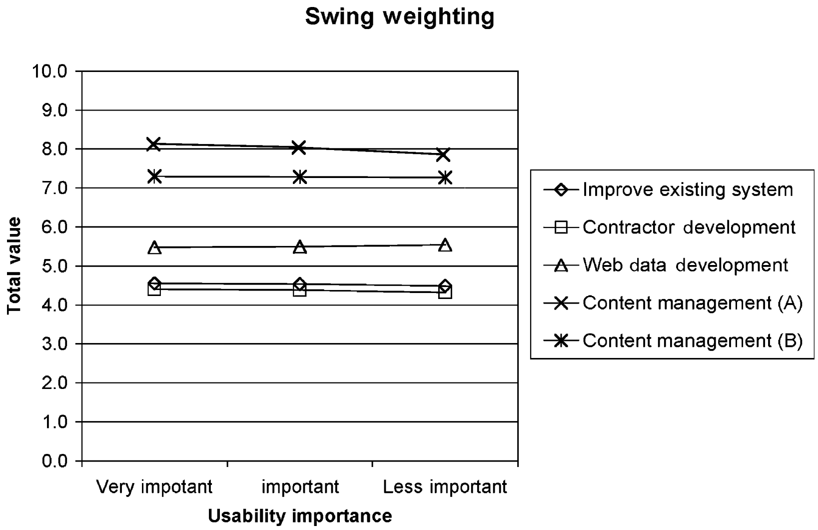


Figure 12.27 Sensitivity analysis for importance of usability.

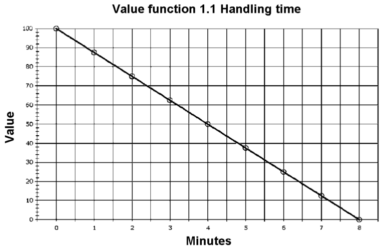
Based on the scoring results, the design team recommended that the department head authorize the purchase of content management software, database software, and development tools from Vendor A so that they could begin development of the proposed system. As the project progressed, they would attempt to integrate capstone students into the development process. This recommendation is based on the fact that content management tools support both information and collaboration with features like file management, discussion, survey response, and e-mail. Vendor A's features provide an advantage over Vendor B with respect to instructor ease of use, flexibility, and integration. The department head, who had been a part of the process from the problem definition phase, agreed with the recommendation and approved the system concept, the software purchase, and the development timeline.

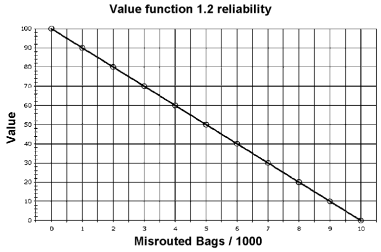
12.14 EXERCISES

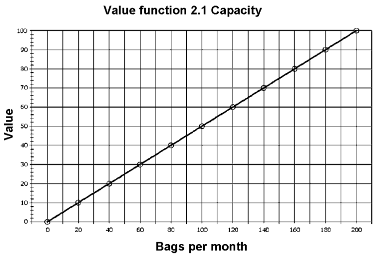
12.1. What important things brought forward from the previous phases in the SDP should a systems engineer review prior to the start of the decision-making step? What should the systems engineer ensure about them prior to moving forward?

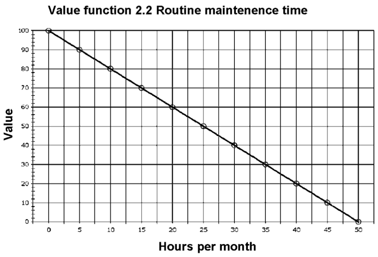
Situation**.** Assume you are the member of a systems design team tasked with designing baggage handling system for the new Baghdad International Airport.

12.2. One measure your team identifies is "Handling Time" or the time to process a bag from the plane to the passenger in the terminal. Describe a method to evaluate this measure for each of the four different ways of scoring alternatives. Identify the strengths and weaknesses of each method.









	Handling time (minutes)	Reliability (misrouted bags/1000)	Capacity (Bags/hrs)	Routine maintenance time (Hrs/Month)
High Tech	2.1	3	190	35
Low Tech	5.3	9	160	45
Mixed	6.2	7	100	25
Manual	7.7	2	75	5

Situation. Your team developed the value curves shown below in previous steps of the process. You have now ducted the scoring of all the alternatives and completed the raw data matrix shown below.

12.3. Given the value curves and the raw data matrix above, develop the value matrix for these candidate solutions.

12.4. Given the weights for each measure in the table below, calculate the value for each candidate solution.

Measure	Weight
Handling time	0.37
Reliability	0.28
Capacity	0.22
Routine maintenance time	0.13

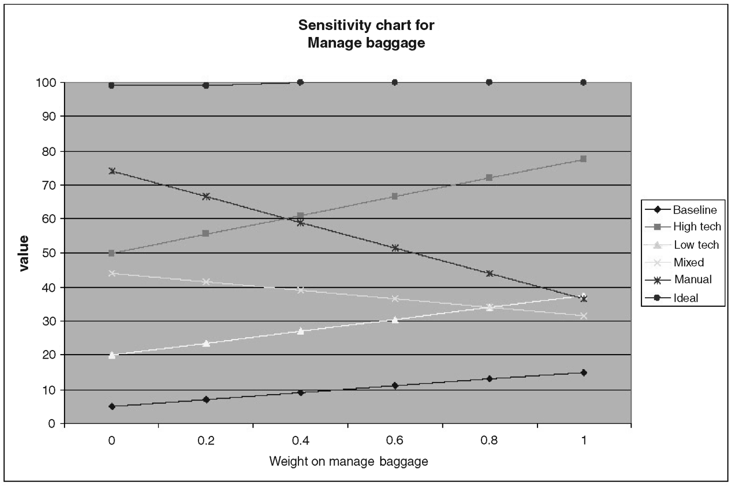
Situation. The Hi-Tech candidate solution calls for the use of very advanced technologies. Some of these technologies are not yet commercially viable (still in development.)

12.5. Is the above statement about the technologies of the Hi Tech candidate solution an example of risk or uncertainty? How would a systems engineer address this in the analysis?

Situation. During the scoring of the candidate solutions, the handling time for the Hi Tech solution and the capacity of the manual solution, though shown above in the raw data matrix as deterministic, were really stochastic.

12.6. Is the above statement about the variability of the measure scores an example of risk or uncertainty? If the handling time for the Hi-Tech solution and the capacity for the manual solution both followed triangle distributions ([1.1, 2.1, 3.1] and [60, 75, 90], respectively), how should a systems engineer address this variability?

12.7. Given the sensitivity chart shown below, how should a systems engineer present the sensitivity of the top-level function "Manage Baggage"? (*Note*: The values in this chart do not completely correspond to the previous numbers in this example as the entire value model is not presented.)



12.8. Use Decision-Focused Transformation to better communicate the value tradeoffs of the nondominated candidate solutions. Is DFT helpful for this problem? Why or why not?

12.9. Given the costs shown below for each candidate solution and the values determined in Exercise 12.4, above, develop the cost versus value graph. Identify the dominated solutions determined in the graph.

Candidate Solution	Cost (millions)
Ideal	$150
High tech	$120
Low tech	$95
Mixed	$75
Manual	$45
Baseline	$15

12.10. Based on all the analysis conducted in the previous exercises for this chapter and especially the cost versus value graph, make a recommendation to a decision maker and justify the recommendation in one or two paragraphs.

12.11. Prepare a one-page executive summary for the decision maker on your analysis of this baggage handling system. Also, develop a top-level outline of the slides you will prepare for the decision briefing that you will present to the decision maker on the analysis conducted above. Assume you only have 20 minutes and you have not presented anything to this decision maker since the approval of your revised problem statement.

REFERENCES

1. Parnell, G. Value-focused thinking. *Methods for Conducting Military Operational Analysis*. Washington, DC: Military Operations Research Society, 2007.
2. Maxwell, D. Decision Analysis: Aiding Insight VII: Decision Analysis Software. In: *OR/MS Today*; Marietta, Georgia, Lionheart Publishing, 2004.
3. Logical Decisions. 2010. Available at http://www.logicaldecisions.com. Accessed August 20, 2010.
4. Kirkwood, C. *Strategic Decision Making: Multiobjective Decision Analysis with Spreadsheets*; Pacific Grove, CA: Duxbury Press, 1997.
5. Ragsdale, C. *Spreadsheet Modeling and Decision Analysis*, 4th ed. Thomson South-Western, 2004.
6. Clemen, R, Reilly, T. *Making Hard Decisions with Decision Tools Suite*. Pacific Grove, CA; Duxbury Press, 2004.
7. Parnell, G, Jackson, J, Burk, R, Lehmkuhl, L, Engelbrecht, J. R&D concept decision analysis: Using alternate futures for sensitivity analysis. *Journal of Multi-Criteria Decision Analysis*, 1999; 8: 119–127.
8. Oracle® Crystal Ball. Denver, CO: Decisioneering, Inc, 2005. Available at. http://www.decisioneering.com. Accessed April 10, 2010.
9. @Risk. Palisade Corporation. Available at http://www.palisade.com, Accessed April 10, 2010.
10. Keeney, R. *Value-Focused Thinking: A Path to Creative Decisionmaking*. Cambridge, MA: Harvard University Press, 1992.
11. Precision Tree by Palidade, http://www.palisade.com/precisiontree/. Accessed August 20, 2010.
12. Dees, R, Dabkowski, M, Parnell, G. Decision-focused transformation of additive value models to improve communication, *Decision Analysis*, 2010; 7: 172–184.
13. Balthazar, T, Dubois, P, Heacock, J, Stoinoff, C, Driscoll, P. A systems perspective on Army ROTC. Department of Systems Engineering Capstone Conference, U.S. Military Academy, West Point, 2007.