

DECISION ANALYSIS



Clear thinking for difficult decisions

REFERENCE



Decision Analysis

Decision Analysis Example

Customer has a memory partition bug in a previous release of the operating system. Your support operation has been given 14 days to provide a fix for this customer problem. The problem has been fixed in a subsequent operating system release. There are a number of ways in which relief can be provided.

KT

Kepner
Tregoe

Decision Analysis

Clarify Purpose

State the Decision

What is the fundamental purpose of this choice?

What specific choice or recommendation needs to be made?

Include:
*Choice word (decide, select...) *Result *1-2 modifiers

Select the best action to take to provide customer with relief from the memory partition bug

Develop Objectives

What short- and long-term benefits or results do you want?

What resources should you use or save?

What restrictions influence this choice?

What minimums or maximums must you meet?

Which objectives need to be clarified?

Consider how time, cost, customers, etc., influence this choice

Be clear and specific Use short statements

Insert New Objective Sort Objectives

Establish measures for
each objective
(as measured by...)

Consider:
Time, speed,
monetary units,
accepted norms,
or other hard numbers.

Classify Objectives

If the objective is Mandatory,

Measurable and Realistic,
label it a MUST.

(Select 'MUST' from drop-down)

Weigh the WANTS

For the other objectives,
what is the relative
importance of each WANT?
(Most important WANT = 10;
Others = 1-10, relative to the '10')

Objectives	Measures	
Prevent problem from being seen in less than 14 days	Number of days	M
Make it no harder to support this machine in the future	Current SLA/policy	M
Minimize engineering effort	Amount of work to be done	4
Minimize risk of introducing new bugs into customer environment	Potential for new bugs based on past record	10
Minimize performance degradation	Number of processing steps added by fix as potential for performance impact	10
Minimize time to relief	Number of days	5
Maximize testing opportunity for fix	Number of test opportunities	7
Maximize customer perception of effective problem resolution	Expected customer reaction	7
Maximize chances of fixing this problem on other customer sites	Ability of fix to be used elsewhere	5

Assess Risks

Identify adverse consequences

Start with the highest performing alternative

Imagine you have implemented this alternative

What risks are associated with this alternative?
What could go wrong, short and long term, if this alternative is chosen?

What are the implications of being close to a MUST limit or threshold?
What information about this alternative might be invalid? What are the implications?

Use "if..., then..." format; e.g., If X happens, then Y is the adverse consequence

Identify adverse consequences for all alternatives that are close to the best performer

Assess the threat

How likely is each adverse consequence? (probability - record the rational; mark each H, M, or L (±))

What level of impact will this adverse consequence have? (seriousness - record the rational; mark each H, M, or L (±))

Make Decision

Make the Choice

Are we willing to accept the risk(s) to gain the benefit of this choice?
Can we manage the risk(s) to an acceptable level?

If yes, pick it
If no, repeat for the next best alternative

Plan and take action to implement the alternative
Plan how you will manage the risk(s)

**Kepner
Tregoe**

Decision Analysis

?

Evaluate Alternatives
Generate Alternatives

What different alternatives are available?

Review Decision Statement and objectives
Consult experts and research many sources
Ask stakeholders who will approve or implement

List alternatives without debate
Use creative thinking techniques
If necessary, combine or design alternatives or consider the status quo

Screen Alternatives Through the MUSTs
Does this alternative meet each MUST objective?

Gather and record factual data
Determine if Go or No Go

Eliminate any No Go alternatives

Use Knowledge and Experience... OR
Which alternative best satisfies each WANT objective?

Examine how each alternative performs against each WANT objective
Mark the best performers with an asterisk ()*

*Each WANT objective should have at least one **
*The alternative with the most * is the best performer*

...Compare Alternatives Against the WANTs
How well does each alternative perform against each WANT objective?

Record supporting data, then for each objective:
Find the best performing alternative and give it a score of 10

Score other alternatives (0-10) relative to the best performer

Multiply objective weights x scores Add for total weighted scores

☐ = Indicate final choice

Total = 381

Alternative 1
Install a full-service release to existing O/S

	Score	Weighted score
This is possible in 14 days	GO	
Standard policy	GO	
Some work to do	4	16
Certainly a chance of bugs	6	60
Full service release adds 1% to processing load	10	100
Will take close to 14 days	3	15
Will be tested multiple times	10	70
They'll have a fix of their own	10	70
Will be available to others	10	50

☐ = Indicate final choice

Total = 300

Alternative 2
Install a single standard workaround

	Score	Weighted score
Can be done in 14 days	GO	
Supported	GO	
Much to do	2	8
Good chance	4	40
Workaround likely to add 4%	6	60
Will take time	6	30
Test at least once	6	42
Their own fix	10	70
Will be available to others	10	50

☒ = Indicate final choice

Total = 385

Alternative 3
Upgrade customer to new version of O/S

	Score	Weighted score
Fix available now	GO	
Supported	GO	
No work to do	10	40
O/S is tested	5	50
New version of OS is equivalent to current processing load	10	100
O/S is ready	8	40
O/S is already tested	10	70
Poor impression	5	35
Will be available to others	10	50

Assess Risks

	Probability	Seriousness
If... full service release does not fix the problem,	L	
then... we are not likely to provide a fix in 14 days		H
If... full service release introduces bugs that were not identified during testing,	H	
then... the customer will be even more unhappy		H
If...		
then...		
If...		
then...		
If...		
then...		

Assess Risks

	Probability	Seriousness
If...		
then...		
If...		
then...		
If...		
then...		
If...		
then...		
If...		
then...		

Assess Risks

	Probability	Seriousness
If... the customer application has not been tested,	L	
then... application performance will be unknown		H
If... the customer application is not certified for the next O/S revision,	L	
then... currently being used on the newer O/S elsewhere in the company)		L
If...		
then...		
If...		
then...		
If...		
then...		

Decision Analysis

[illegible]