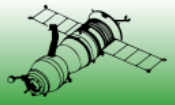
INCOSE International Symposium
Las Vegas, NV
June 30 - July 3, 2014

When “Yes” is the Wrong Answer

Richard Beasley, Andy J Nolan and Andrew C. Pickard

INCOSE 2014 International Symposium


The good, the bad and the ugly

Introduction

- System Engineering's value comes from doing effective pre-work to:
  - Get a full understanding of situation
  - Drive informed decisions, leading to effective outcomes
  - Avoid later, expensive rework
- There are many barriers – and one is the way we review and lead our projects
- Many questions asked regarding progress and status prevent proper use of Systems Engineering because the expected / desired answer is "yes"
  - Better answer would be "no" or "not yet"
- We give examples and illustrate the problems implicit often due to an expectation of linear progress

Good

Example Questions

- Are your requirements complete?
- Do you understand all of the interfaces?
- It's only a small change – can I skip the analysis and test?
- Have you mitigated all the risks?
- Have you used stage gates/independent review?
- Can you improve the system by changing one part?
- Do you think the customer is an idiot?

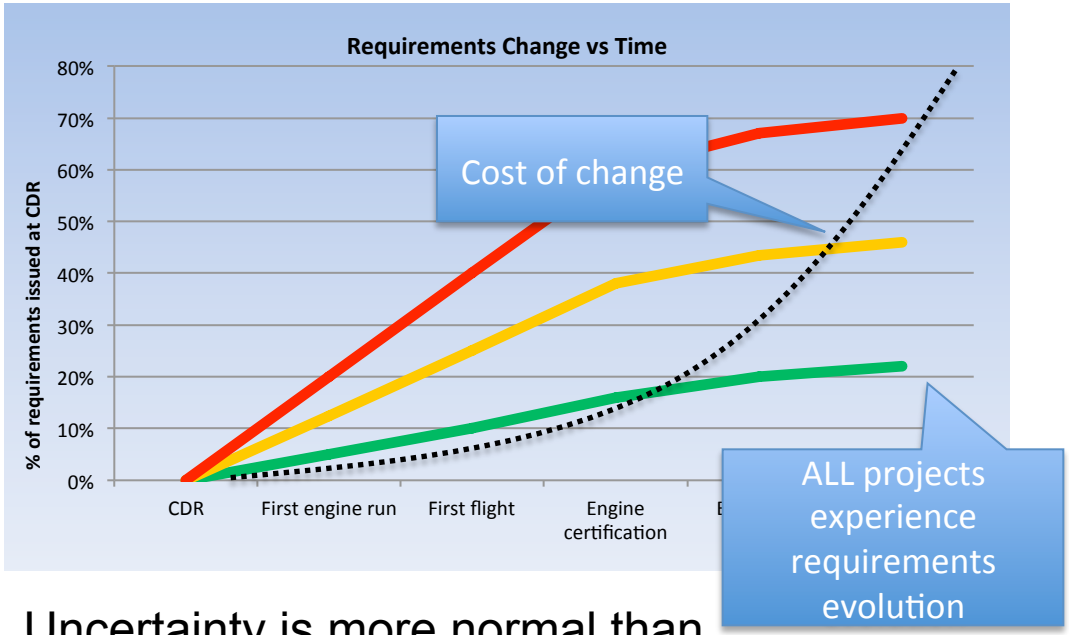
Are your requirements complete?

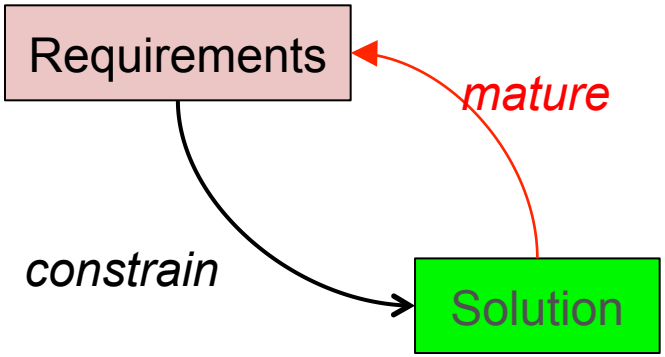
INCOSE International Symposium
Las Vegas, NV
June 30 - July 3, 2014

At first glance, an entirely sensible question. But…

Of 10 projects measured at RR, all had requirements change:

- Wicked problems
- Emergent requirements
- Customers never know all their needs



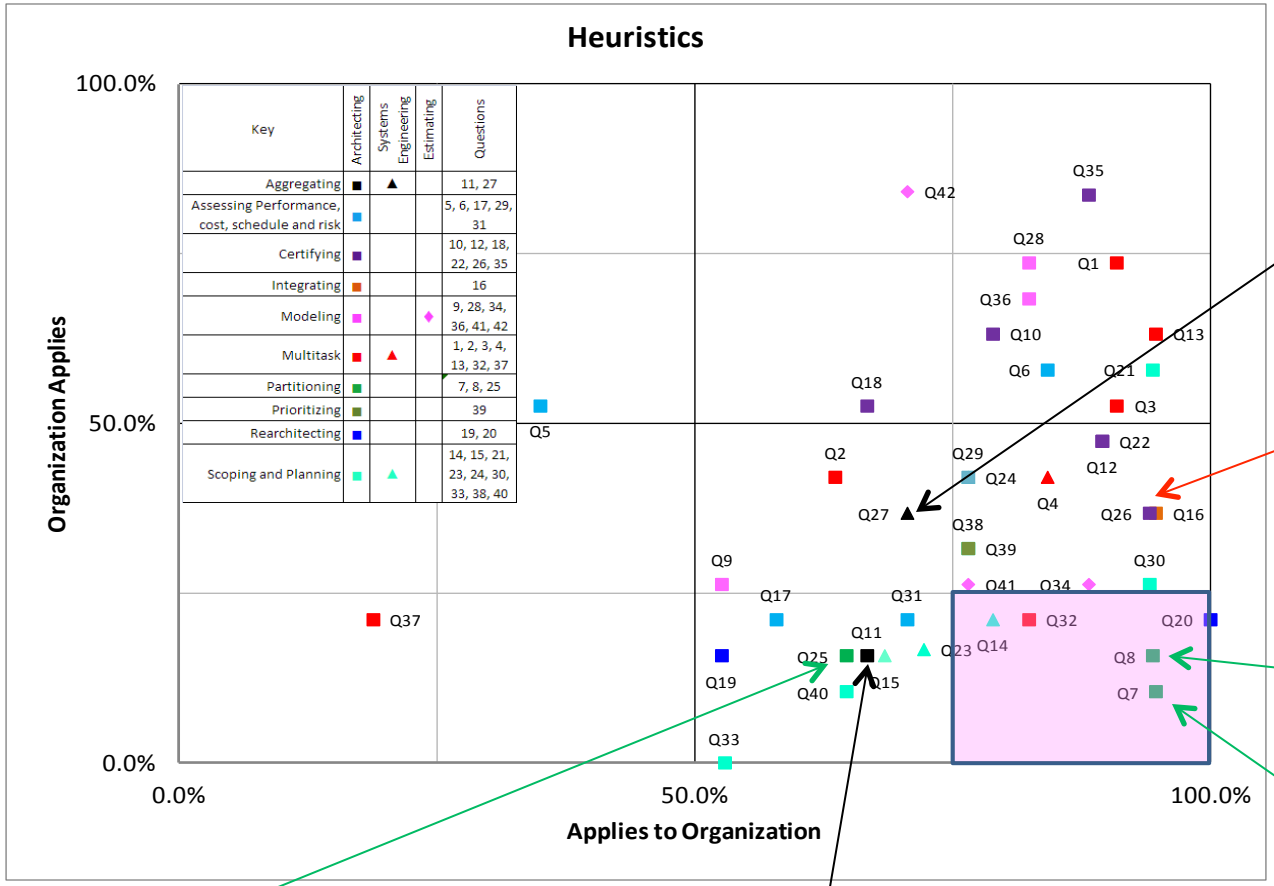


Uncertainty is more normal than certainty. Assuming you have complete requirements and ignoring the uncertainty will increase costs. Requirements uncertainty management has a 100:1 ROI

Bad

Do you understand all of the interfaces?

INCOSE International Symposium
Las Vegas, NV
June 30 - July 3, 2014



Q27 Modularity - To build a durable system, keep integration low between modules

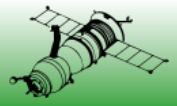
Q16 The greatest leverage in system architecting is at the interfaces; the greatest dangers are also at the interfaces

Q8 Be prepared for reality to add a few interfaces of its own

Q25 Organize personnel tasks to minimize the time individuals spend interfacing

Q11 Choose a configuration with minimal communications between the subsystems

Q7 It is inadequate to architect up to the boundaries or interfaces of a system; one must architect across them


Ugly

It's only a small change – can I skip the analysis and test?

- Small changes need risk assessment and mitigation -
  - It is too easy to make invalid assumptions
- Sweeping assumptions can be removed by clear thinking and a realistic assessment of risk
  - showing "simple" changes are not always simple
- If it is simple, then SE analysis will quickly show it to be so
- Question often driven by an implicit need for the change to be simple
- Better question – "is it a simple change, and how do you know?"

Ass u me **Assume**

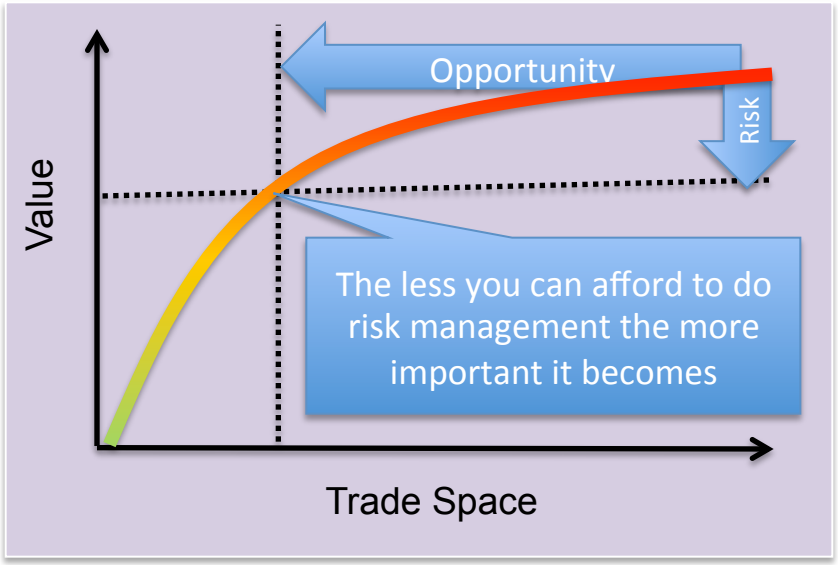
Good

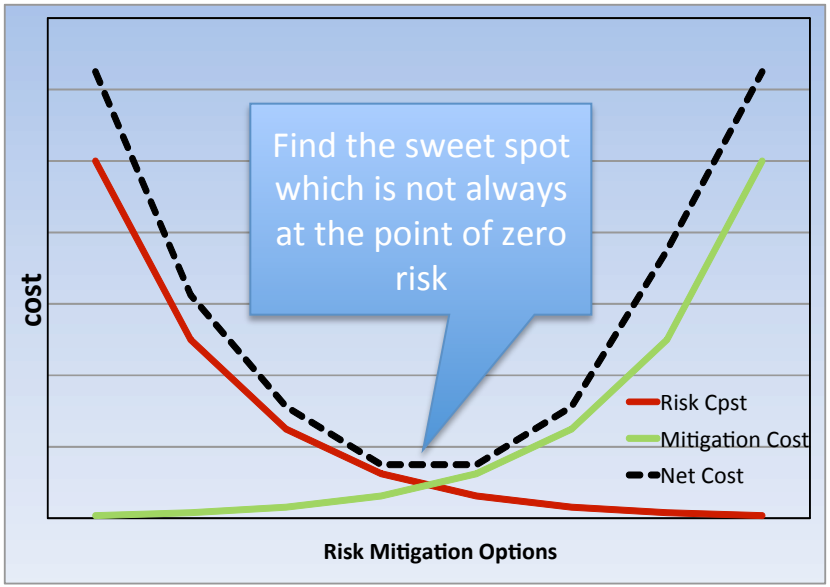
Have you mitigated all the risks?

INCOSE International Symposium
Las Vegas, NV
June 30 - July 3, 2014

At first glance, an entirely sensible question. But…

- Zero risk is not the most cost effective position to be in
- Taking risks for greater opportunity is a normal part of systems engineering.

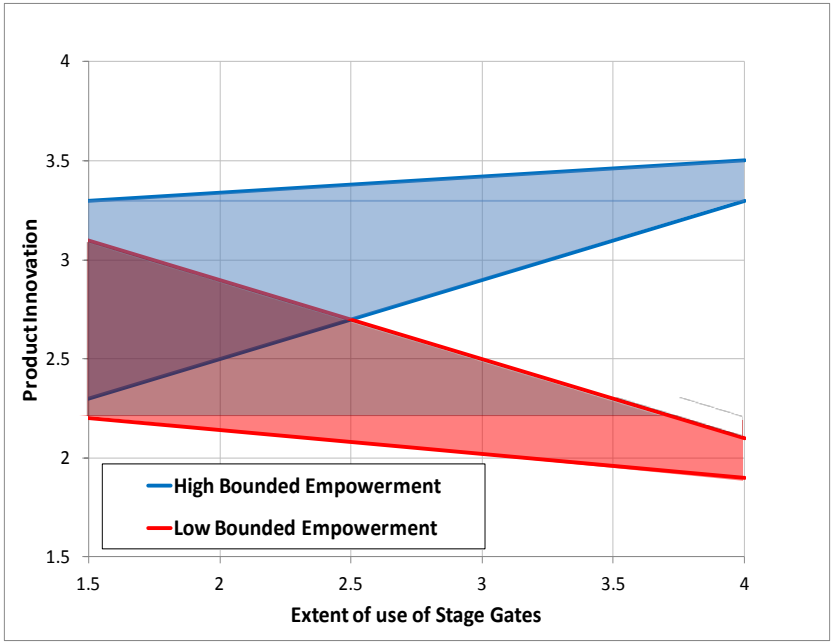




Risk management is not there to remove all risk but to help you take informed risks.

Have you used stage gates/ independent review?

- **Effectiveness of reviews is driven by culture and attitude**
- **Right attitude** – an opportunity to help avoid problems later
- **Wrong attitude** - an inspection - if problems are hidden / not found by the team then we passed
- **Do you trust team and review, or try to inspect quality in?**



From Hull, Frank, 2013 - Society of Concurrent Product Development - see http://www.scpdnet.org/

The impact of independent review depends on the level of high bounded empowerment in the team

Good

Have you got a solution yet?

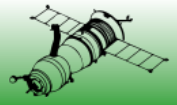
INCOSE International Symposium
Las Vegas, NV
June 30 - July 3, 2014

Where Found \ Where Should Have Been Found	1 - Setting or Change of Requirements	2 - Requirements Review	3 - Design Analysis	4 - Design Review	5 - Supply Chain Issues	6 - Manufacture	7 - Component Test	8 - Module Assembly	9 - Module Test	10 - Development Engine Assembly	11 - Engine Development Test	12 - First Article Inspection	13 - Engine Certification Test	14 - Production Engine Assembly	15 - Production Engine Test	16 - Flight Test	17 - In Service	Cost Weight	Cost iff found at right stage	Actual cost
1 - Setting or Change of Requirements	37	1		4														1	56	42
2 - Requirements Review	2	3																1	21	5
3 - Design Analysis	3	1	15							1								1	69	20
4 - Design Review	2	5	4	10														1	63	21
5 - Supply Chain Issues	1	1		6	5													2	12	26
6 - Manufacture	1	3	5	2	1	10												2	24	44
7 - Component Test		1	11	3			2											5	20	85
8 - Module Assembly		5	2	3				1										10	10	110
9 - Module Test									2									10	30	20
10 - Development Engine Assembly	1		3	4						1								25	50	225
11 - Engine Development Test	1		13	13		1			1	1	22							25	950	1300
12 - First Article Inspection				1														25	0	25
13 - Engine Certification Test		1		1														50	50	100
14 - Production Engine Assembly				1							1							100	0	200
15 - Production Engine Test				2							4							100	0	600
16 - Flight Test	1	1	2	1			1				3		1					100	0	1000
17 - In Service	7		14	16		1	1				8						12	1000	12000	59000
Total Escapes	19	18	54	53	1	2	2	0	1	1	16	0	1	0	0	0	12	Total:	13355	62823
Total	56	22	69	67	6	12	4	1	3	3	38	0	1	0	0	0	12	Cost Ratio:		470%

Key:
>= 20
10 to 19
5 to 9
2 to 4
1

"Where found/where should have been found" analysis for 294 design modifications performed by 9 different design groups

- By far the majority of the problems with the original designs should have been detected before the design solution was finalized
- What is needed: A better understanding of requirements (what is the function the design is intended to perform, and in what environment), a more rigorous analysis of the capability of the design and a design review that is much more than a "check-box" exercise
- A better question to receive a "yes" response to is "Do you understand the problem (that the design solution needs to address) yet?" The probability of a good solution will increase if the problem understanding is mature enough for solution work to commence.


Ugly

Are you going to get it right first time?

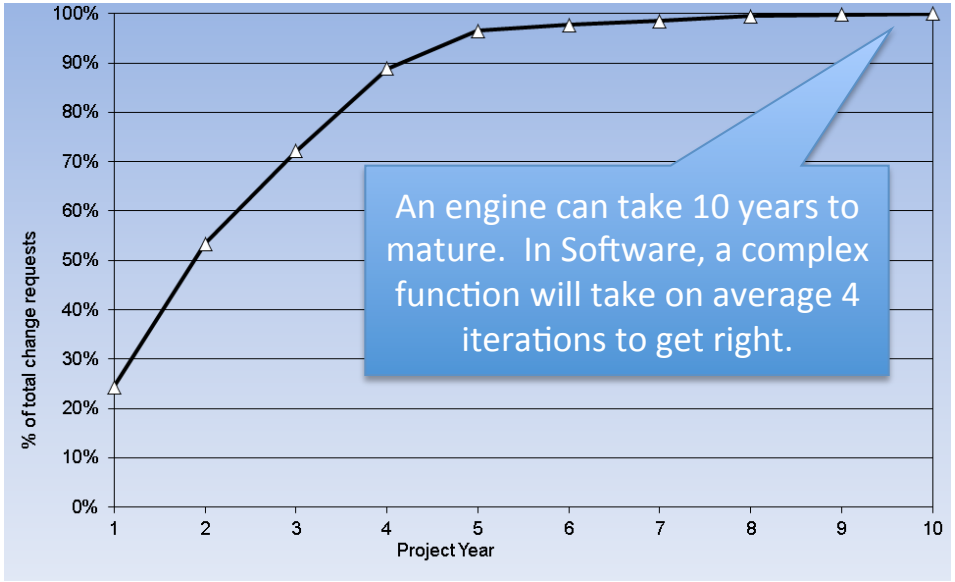
At first glance, an entirely sensible goal. But…

This will drive the behaviour of

•Planning for "failure" becomes politically incorrect.
•Product evolution and certification are combined into a single pass "heavy weight" process
•Contingency, mitigation and backup plans are removed

This then causes:

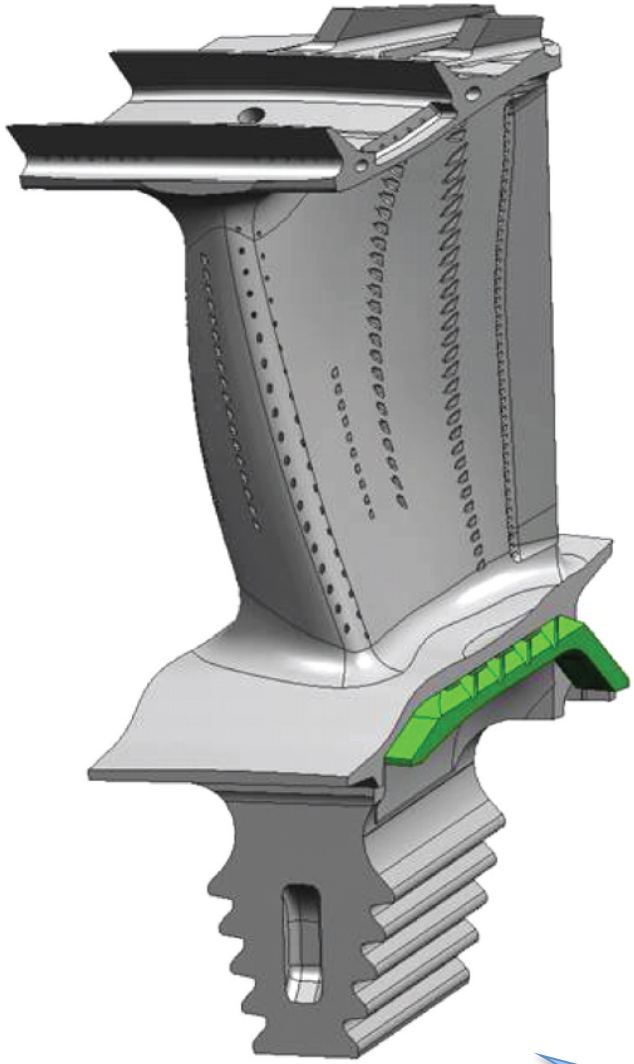
•late changes to become a surprise and more expensive



Complex projects need time to mature. Rework is only bad if you had not planned for it! When you plan for rework, plan to rework it as soon as possible in the most effective way.

Bad

Can you improve the system by changing one part?



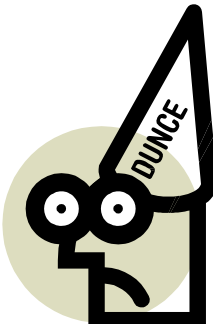
- ormed by
- shroudless
- tem.

Ugly

Do you think the customer is an idiot?

INCOSE International Symposium
Las Vegas, NV
June 30 - July 3, 2014

- It is easy to be “frustrated” with our customers
- We have to manage our expectations
  - Customer’s situation change as much as our situation
  - Expecting complete information leads to an abdication of our design team responsibilities
  - Complete information would be “over-constraining” – we’d be given solutions to draw, not problems to solve



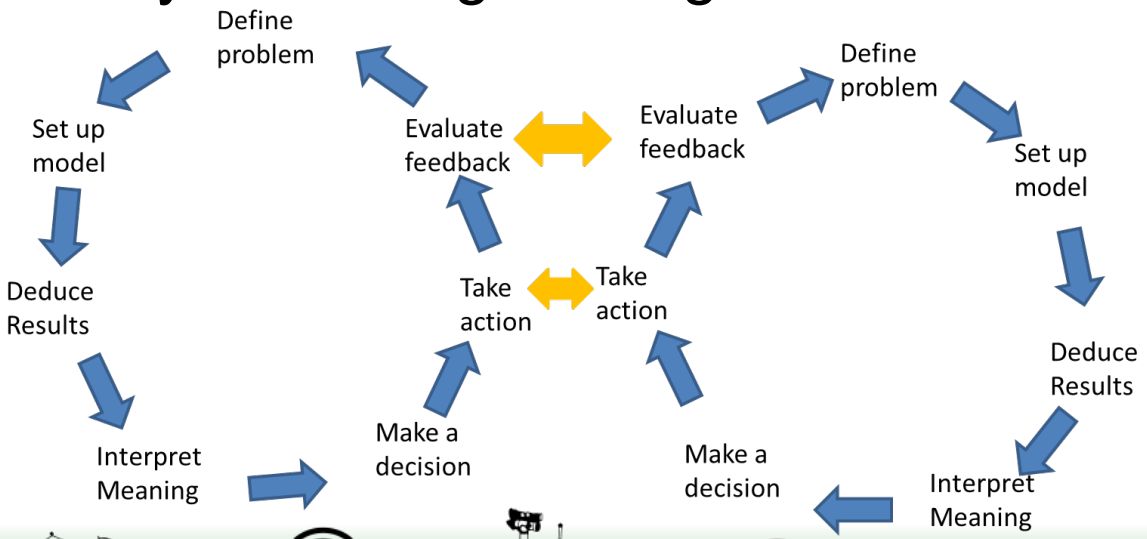
Even thinking this can block the ability the understand the customer and the situation

  - Soft Systems (Checkland) emphasizes understanding world view of all different stakeholders
  - Basic Emotional intelligence shows that “red” emotions (negativity leads to critical) creates inability to be aware of their situation

Good

Pre-Work, not Rework

- Pre-work is NOT starting early - it focuses on removing uncertainty/ increasing understanding
- Program plans need
  - Iteration to exploit the understanding achieved
  - Get timely understand to support decisions
- Without planned in iteration in the plan then the potential value of Systems Engineering cannot be realised.



Adapted from Blockley, D and Godfrey, P, Doing it Differently – 2000

Good

Who's your hero?

INCOSE International Symposium
Las Vegas, NV
June 30 - July 3, 2014

- "Every day some new fact comes to light – some new obstacle which threatens the gravest obstruction. I suppose this is the reason which makes the game so well worth playing"

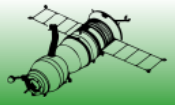
  Robert Falcon Scott, Polar explorer 1868-1912

  Second expedition leader to reach the South Pole

- "Adventure is just bad planning"

  Roald Amundsen, Polar explorer 1872-1928

  First expedition leader to reach the South Pole


Good

Heuristics

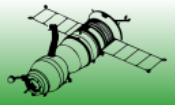
Showing progress on a linear path is not necessaril
might be in the

Recognizing uncertainty is the first
step to certainty and to success.
ainty means failure
ore likely

The customer may well not be right, but their position is valid from their (current) point of view and should be respected

It's not enoug
Systems Engineering; you must plan to do something with what you find doing Systems Engineering

a project cheaper by doing less; you make it cheaper by doing more of the right things.


Bad

Conclusions

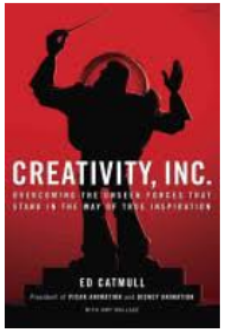
- **Many typical "management" questions**
  - Reinforce linear planning / tendency to jump to component solutions
  - Avoid valuing the identification and management of uncertainty.
- **The purpose of Systems Engineering is to improve the probability of a successful outcome to a complex/messy problem.**
  - It does this by looking for understanding of the problem, and
  - Uses that understanding to inform work to define the solution.
- **Discovering uncertainty is the first step towards certainty. "Yes" expectation hides that uncertainty**
- **Plan to look to identify, and then reduce uncertainties; adapting to what is found**
- **This must be a key common understanding between Systems Engineers and Program Managers.**

Finally…. "Have you made it simple?"

"When you distil a complex idea to a T-shirt slogan

- You risk giving the illusion of understanding
- In the process you sap the idea of its power

You end up with something that is easier to say, but not connected to behaviour"

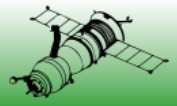


Ed Catmulll
Creativity, Inc.
Overcoming the Unseen forces that Stand in the Way of True Inspiration
2014

A better question –

"Is it well enough understood to give clear communication?"

Hopefully we have – but any questions anyway?


Good

Homework

- Do you know of any other typical questions to which the answer “Yes” is the wrong answer?