ECED3901 Design Methods II

LECTURE #2: DESIGN PROCESS

Colin O'Flynn

What are we covering?

Part #1 – Designs

What makes a good product?

Good Design?

Bad Design?

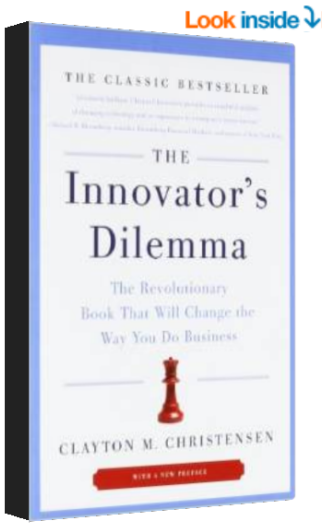
https://openlab.citytech.cuny.edu/artlife/2014/05/12/bad-design1/

Design Pitfalls

Step 1: Listen to Customers.

Step 2: Ignore Your Customers.

(The Following Loosely Adapted From)



Flip to back

The Innovator's Dilemma: The Revolutionary Book That Will Change the Way You Do Business Paperback – October 4, 2011

by Clayton M. Christensen (Author)

172 customer reviews

#1 Best Seller in Customer Relations

See all 7 formats and editions

Paperback
$13.50

38 Used from $6.70
69 New from $8.19

"Absolutely brilliant. Clayton Christensen provides an insightful analysis of changing technology and its importance to a company's future success."
—Michael R. Bloomberg

"This book ought to chill any executive who feels bulletproof —and inspire entrepreneurs aiming their guns."

Hype is turned up to 11 so read with caution, but has some interesting material anyway

Product Types: Iterative

Product Types: Disruptive

Examples of Disruptive Products

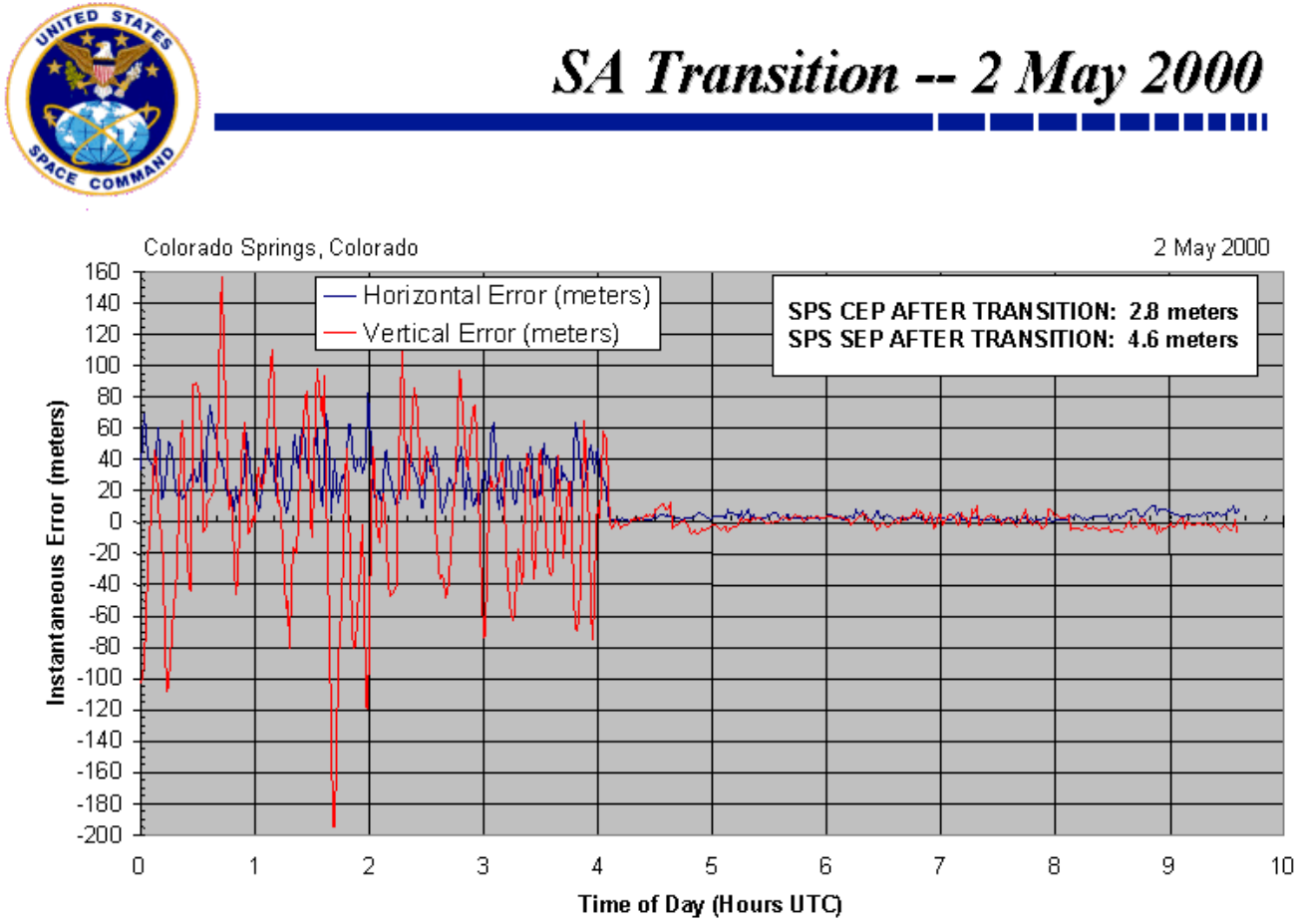
Wikipedia → Classic Encyclopedia

Telephones → Telegraph

Digital Camera → Film Camera

Timing for Disruptive Products

Example: GPS network turned Selective Availability Off



Sidebar: What is a Product?

The first public paper on turbo codes was "*Near Shannon Limit Error-correcting Coding and Decoding: Turbo-codes*".[1] This paper was published 1993 in the Proceedings of IEEE International Communications Conference. The 1993 paper

Turbo codes were so revolutionary at the time of their introduction that many experts in the field of coding did not believe the reported results. When the performance was confirmed a small revolution in the world of coding took place

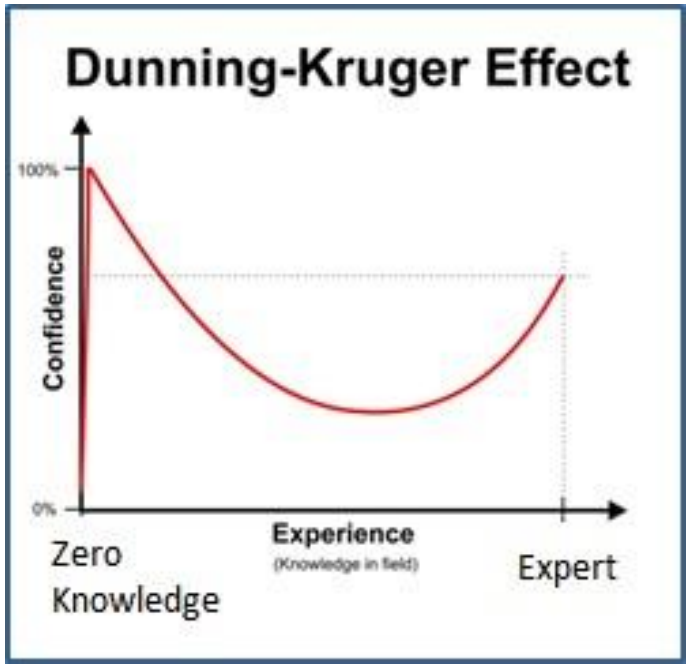
Impractical to implement when first developed by Gallager in 1963,[6] Gallager's LDPC codes were forgotten until Gallager's work was discovered in 1996.[7] Turbo codes, another class of capacity-approaching codes discovered in 1993, became the coding scheme of choice in the late 1990s, used for applications such as the Deep Space Network and satellite communications. However, in the last few years, the advances in low-density parity-check codes have seen them surpass turbo codes in terms of error floor and performance in the higher code rate range, leaving turbo codes better suited for the lower code rates only.[8]

Source: http://en.wikipedia.org/wiki/Turbo_code and http://en.wikipedia.org/wiki/Low-density_parity-check_code

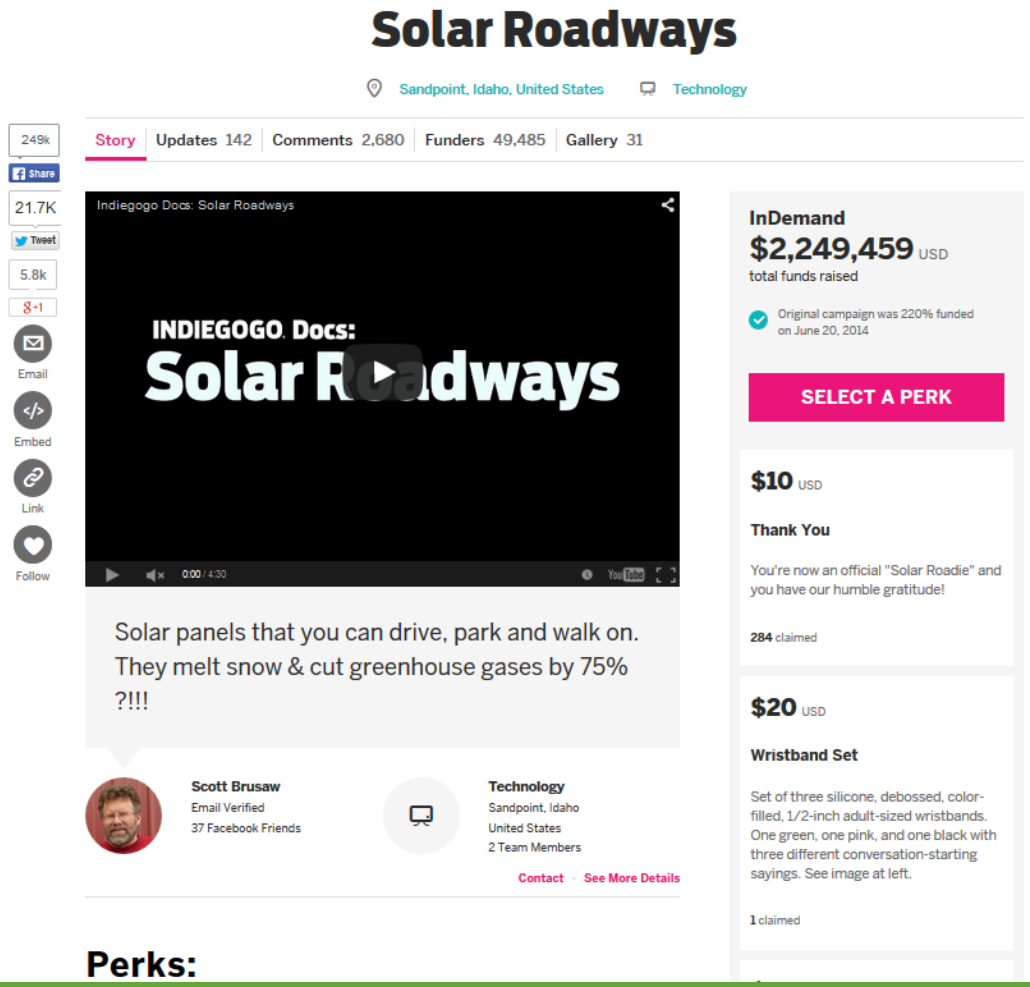
You are Probably Wrong

Dunning–Kruger effect

“Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments”

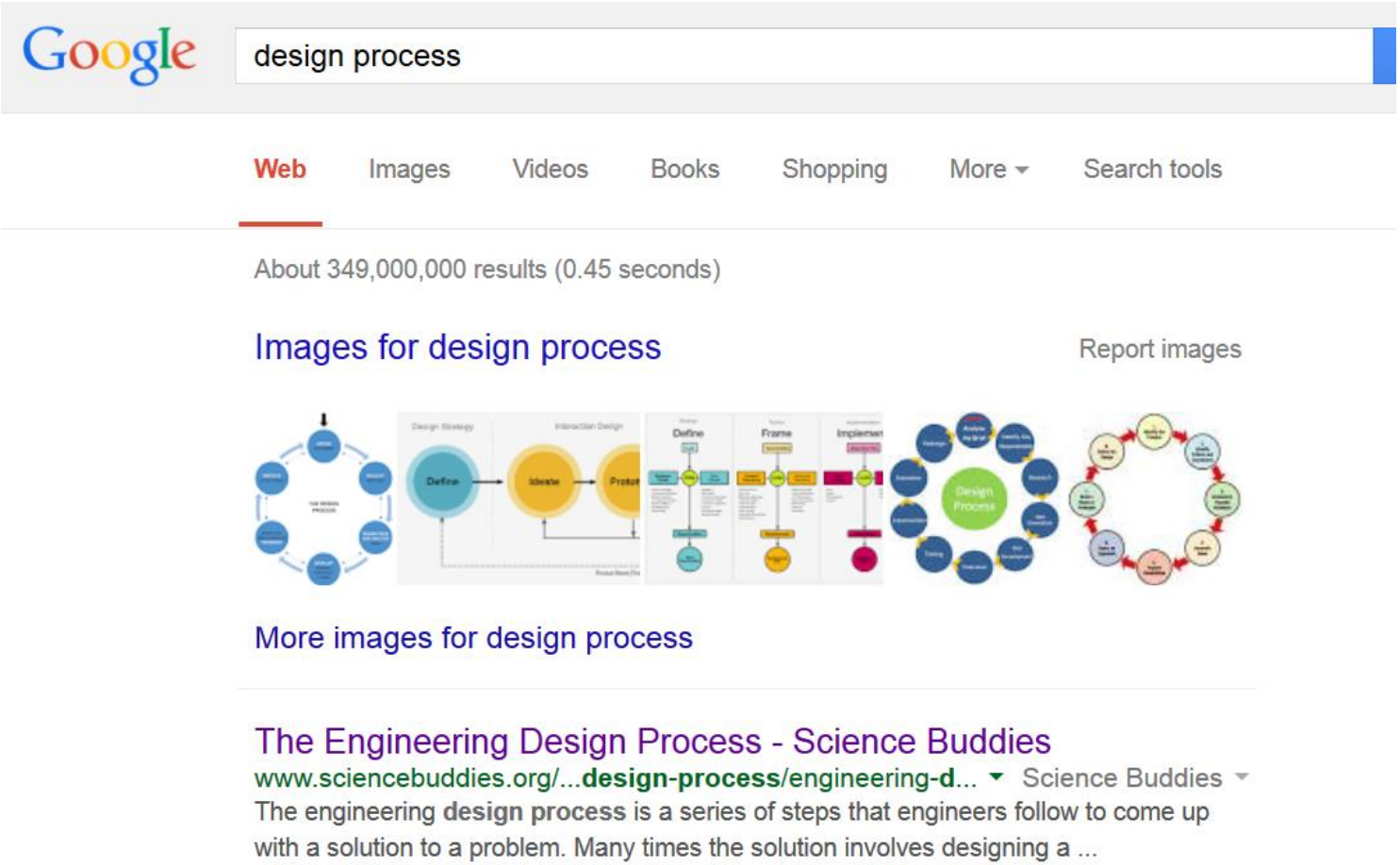


Impossible Products

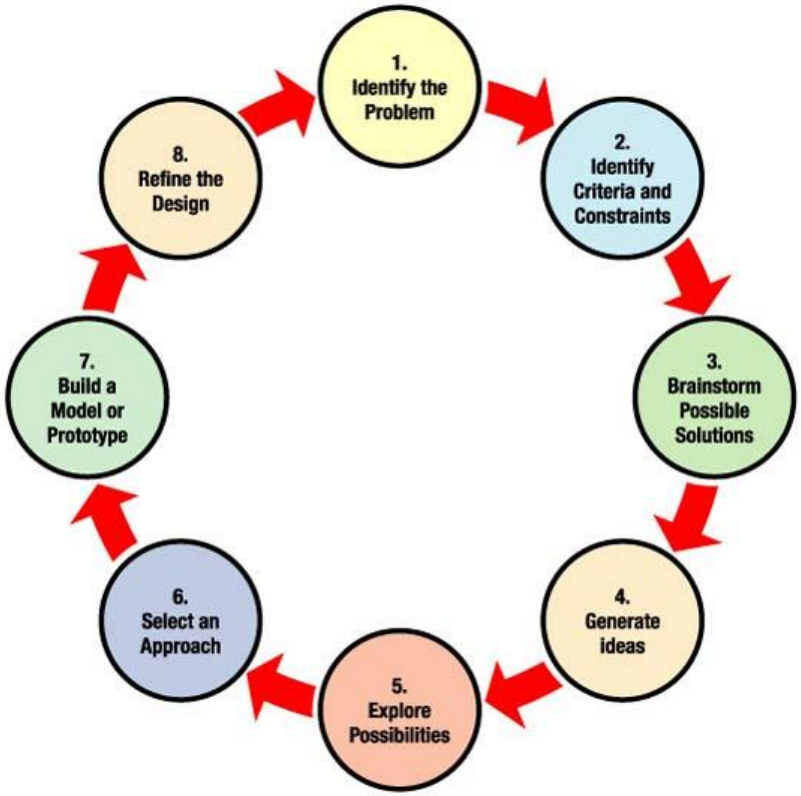


Colin O'Flynn

“Design Process”



“Design Process”

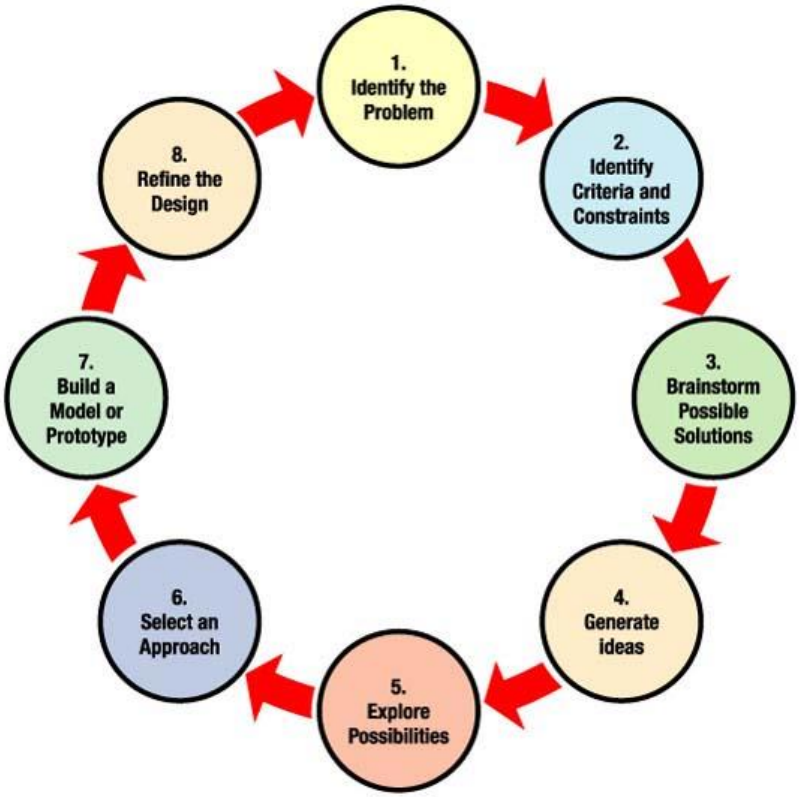


Source: http://www.nasa.gov/audience/foreducators/plantgrowth/reference/Eng_Design_5-12_prt.htm

Realistic Design Process

- Don’t be afraid to iterate frequently
- Plan on making many loops – exploring one aspect might require a run through the entire “design process” – that run lasting 45 mins

Design Process



Idea Generation

Conceptual Design

Detailed Design

Prototyping

Refinement / Ramp-Up

Source: http://www.nasa.gov/audience/foreducators/plantgrowth/reference/Eng_Design_5-12_prt.htm

Idea Generation

Problem statement
Scope of solution
Functions Required
Specifications
Constraints (physical, funding, legal)
Approximate Scheduling
Brainstorming

Conceptual Design

Concepts Proposed

Proof-of-Concept Prototypes

Manufacturing Constraints explored

Possible early Customer Feedback

Detailed Design

Electrical design

Mechanical design

Supply-chain validated (i.e. are parts End-of-Life (EOL)?)

Prototyping (again)

Documentation started

Prototyping

‘Real’ prototypes (approaching full-featured)

Possible issues identified, possible variants tested

i.e.: We have choice of four connectors, which is most reliable? Make PCB with each connector, put through testing

Closer discussions with eventual manufacture, Design For Manufacturing (DFM) considerations raised/tested/fixed.

Test/QA plan formalized

Refinement / Ramp-Up

Small production run (if applicable)

Full production run (if applicable)

Finalize specs, datasheet

Part #2 – Design Methods

Stage-Gate Method

Colin O’Flynn

Agile Development

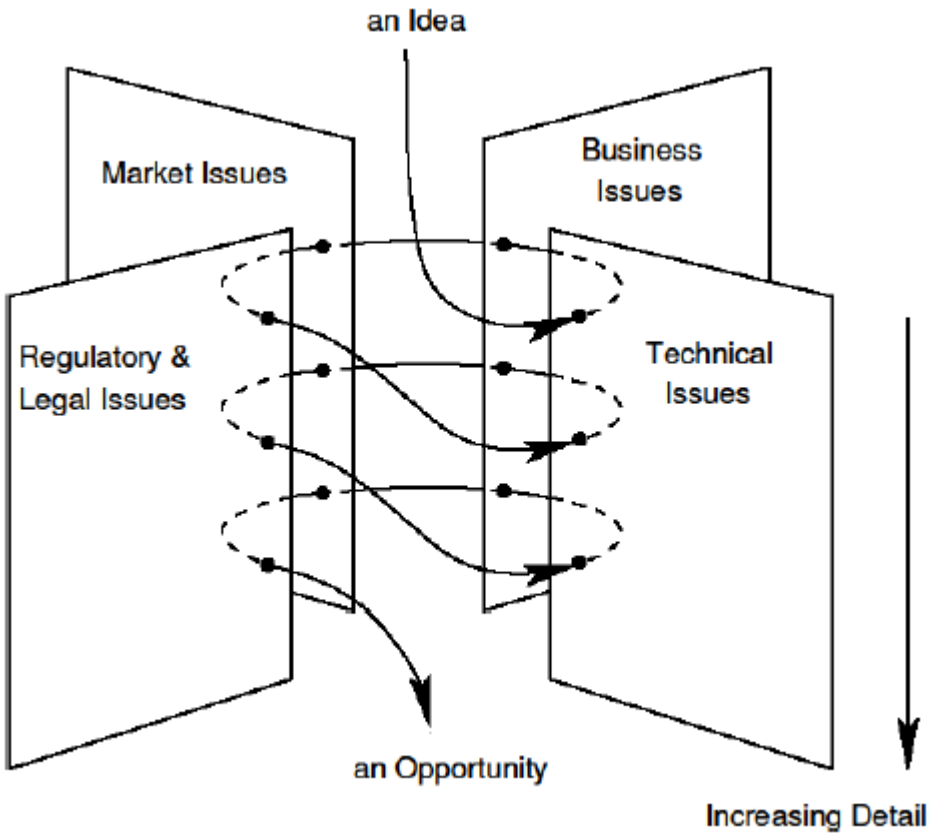
- Normally used for software development
- Holds “Sprints” during which a potential product/version is generated
- Sprints are short time-frames (weeks-month)
- Customer shown this version, possible feedback if not what they are looking for

SCRUM

See http://www.slideshare.net/jaaronfarr/scrum-agile-for-everyone

Part #3 - Idea Generation

Opportunity Helix



Brainstorming

- Focus on quantity
- Add unusual / impossible ideas
- Combine / improve ideas
- Take a break between sessions to mull over ideas!

Proof of Concept Testing (Again)

Use with any of the previous methods…

Summary

- Good design achieves the goal, determining *what* the goal is might be much harder than just asking customers (but where you should start).
- Avoid feature creep & desire to use or invent new technologies unless they directly achieve the goal.
- Use proof-of-concept (PoC) testing early and often, sometimes before goal has been fully defined.
- Disruptive products/innovations often invent entirely new markets or open markets to new customers. They take longer to become profitable (normally) and more likely to fail.